(12)特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局

(43) 国際公開日
2001年10月18日 (18.10.2001)

PCT

(10) 国際公開番号
WO 01/78299 A1

(51) 国際特許分類[7]: H04L 9/00, G06F 17/60, G11B 20/10, 20/12

(21) 国際出願番号: PCT/JP01/02929

(22) 国際出願日: 2001年4月4日 (04.04.2001)

(25) 国際出願の言語: 日本語

(26) 国際公開の言語: 日本語

(30) 優先権データ:
特願2000-105329 2000年4月6日 (06.04.2000) JP
特願2000-179692 2000年6月15日 (15.06.2000) JP
特願2000-317803 2000年10月18日 (18.10.2000) JP

(71) 出願人 (米国を除く全ての指定国について): ソニー株式会社 (SONY CORPORATION) [JP/JP]; 〒141-0001 東京都品川区北品川6丁目7番35号 Tokyo (JP).

(72) 発明者; および
(75) 発明者/出願人 (米国についてのみ): 石黒隆二 (ISHIGURO, Ryuji) [JP/JP]. 大澤義知 (OSAWA, Yoshitomo) [JP/JP]. 大石丈於 (OISHI, Tateo) [JP/JP]. 浅野智之 (ASANO, Tomoyuki) [JP/JP]. 光澤 敦 (MITSUZAWA, Atsushi) [JP/JP]; 〒141-0001 東京都品川区北品川6丁目7番35号 ソニー株式会社内 Tokyo (JP).

(74) 代理人: 小池 晃, 外 (KOIKE, Akira et al.); 〒105-0001 東京都港区虎ノ門二丁目6番4号 第11森ビル Tokyo (JP).

(81) 指定国 (国内): CA, CN, IN, KR, RU, SG, US.

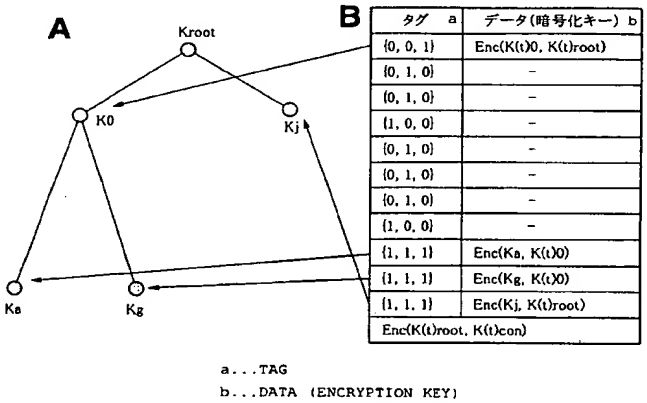
(84) 指定国 (広域): ヨーロッパ特許 (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR).

添付公開書類:
— 国際調査報告書

2文字コード及び他の略語については、定期発行される各PCTガゼットの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

(54) Title: INFORMATION PROCESSING SYSTEM AND METHOD

(54) 発明の名称: 情報処理システム及び方法



(57) Abstract: An effective key block (EKB) used for key distribution structure of tree structure is re-structured to a simplified two-branch or multi-branch tree in which a decodable end node or leaf is at the lowest layer so as to create a re-structured layered tree according to only a key corresponding to the node or leaf of the re-structured layered tree. A tag as identification data about the tree position of an encryption key stored in the EKB is stored. In the tag, not only the position identification but data for judging if encryption key data is present in the EKB is stored. The amount of data in the EKB is greatly reduced and the decoding by a device is simplified. Thus, providing an information processing system and method enabling reduction of the amount of data in an Effective key block (EKB) used for an encryption key structure of tree structure.

[続葉有]

WO 01/78299 A1

(57) 要約:

ツリー構造のキー配布構成に用いる有効化キーブロック（ＥＫＢ）を、復号可能な末端ノード又はリーフを最下段として簡略化した２分岐または多分岐型ツリーを再構築して、再構築階層ツリーのノード又はリーフに対応するキーのみに基づいて生成する。さらに、ＥＫＢに格納した暗号化キーのツリー位置の識別データとしてのタグを格納する。タグは、位置識別のみならず、ＥＫＢ内の暗号化キーデータの有無を判別するデータを格納した構成とした。ＥＫＢの大幅なデータ量削減が実現するとともに、デバイスでの復号処理も簡易化される。すなわち、ツリー構造の暗号化キー構成に用いる有効化キーブロック（ＥＫＢ）のデータ量の削減を可能とした情報処理システム及び方法を実現する。

明細書

情報処理システム及び方法

技術分野

本発明は、情報処理システム、情報処理方法、及び情報記録媒体、並びにプログラム提供媒体に関し、特に、暗号処理を伴うシステムにおける暗号処理鍵を配信するシステム及び方法に関する。特に、木構造の階層的鍵配信方式を用い、さらに、階層的鍵配信ツリーを配信デバイスに応じて再構築して配信キーブロックに含まれるデータ量を削減することにより、配信メッセージ量を小さく抑えて、コンテンツキー配信、あるいは各種鍵の更新の際のデータ配信の負荷を軽減し、かつデータの安全性を保持することを可能とする情報処理システム、情報処理方法、及び情報記録媒体、並びにプログラム提供媒体に関する。

背景技術

昨今、ゲームプログラム、音声データ、画像データ等、様々なソフトウェアデータ（以下、これらをコンテンツ（Content）と呼ぶ）を、インターネット等のネットワーク、あるいはＤＶＤ、ＣＤ等の流通可能な記憶媒体を介しての流通が盛んになってきている。これらの流通コンテンツは、ユーザの所有するＰＣ（Personal Computer）、ゲーム機器によってデータ受信、あるいは記憶媒体の装着がなされて再生されたり、あるいはＰＣ等に付属する記録再生機器内の記録デバイス、例えばメモリカード、ハードディスク等に格納されて、格納媒体からの新たな再生により利用される。

ビデオゲーム機器、ＰＣ等の情報機器には、流通コンテンツをネットワークから受信するため、あるいはＤＶＤ、ＣＤ等にアクセスするためのインタフェースを有し、さらにコンテンツの再生に必要となる制御手段、プログラム、データの

メモリ領域として使用されるＲＡＭ、ＲＯＭ等を有する。

音楽データ、画像データ、あるいはプログラム等の様々なコンテンツは、再生機器として利用されるゲーム機器、ＰＣ等の情報機器本体からのユーザ指示、あるいは接続された入力手段を介したユーザの指示により記憶媒体から呼び出され、情報機器本体、あるいは接続されたディスプレイ、スピーカ等を通じて再生される。

ゲームプログラム、音楽データ、画像データ等、多くのソフトウェア・コンテンツは、一般的にその作成者、販売者に頒布権等が保有されている。したがって、これらのコンテンツの配布に際しては、一定の利用制限、すなわち、正規なユーザに対してのみ、ソフトウェアの使用を許諾し、許可のない複製等が行われないようにする、すなわちセキュリティを考慮した構成をとるのが一般的となっている。

ユーザに対する利用制限を実現する１つの手法が、配布コンテンツの暗号化処理である。すなわち、例えばインターネット等を介して暗号化された音声データ、画像データ、ゲームプログラム等の各種コンテンツを配布するとともに、正規ユーザであると確認された者に対してのみ、配布された暗号化コンテンツを復号する手段、すなわち復号鍵を付与する構成である。

暗号化データは、所定の手続きによる復号化処理によって利用可能な復号データ（平文）に戻すことができる。このような情報の暗号化処理に暗号化鍵を用い、復号化処理に復号化鍵を用いるデータ暗号化、復号化方法は従来からよく知られている。

暗号化鍵と復号化鍵を用いるデータ暗号化・復号化方法の態様には様々な種類あるが、その１つの例としていわゆる共通鍵暗号化方式と呼ばれている方式がある。共通鍵暗号化方式は、データの暗号化処理に用いる暗号化鍵とデータの復号化に用いる復号化鍵を共通のものとして、正規のユーザにこれら暗号化処理、復号化に用いる共通鍵を付与して、鍵を持たない不正ユーザによるデータアクセスを排除するものである。この方式の代表的な方式にＤＥＳ（データ暗号標準：Deta encryption standard）がある。

上述の暗号化処理、復号化に用いられる暗号化鍵、復号化鍵は、例えばあるパ

スワード等に基づいてハッシュ関数等の一方向性関数を適用して得ることができる。一方向性関数とは、その出力から逆に入力を求めるのは非常に困難となる関数である。例えばユーザが決めたパスワードを入力として一方向性関数を適用して、その出力に基づいて暗号化鍵、復号化鍵を生成するものである。このようにして得られた暗号化鍵、復号化鍵から、逆にそのオリジナルのデータであるパスワードを求めることは実質上不可能となる。

また、暗号化するときに使用する暗号化鍵による処理と、復号するときに使用する復号化鍵の処理とを異なるアルゴリズムとした方式がいわゆる公開鍵暗号化方式と呼ばれる方式である。公開鍵暗号化方式は、不特定のユーザが使用可能な公開鍵を使用する方法であり、特定個人に対する暗号化文書を、その特定個人が発行した公開鍵を用いて暗号化処理を行う。公開鍵によって暗号化された文書は、その暗号化処理に使用された公開鍵に対応する秘密鍵によってのみ復号処理が可能となる。秘密鍵は、公開鍵を発行した個人のみが所有するので、その公開鍵によって暗号化された文書は秘密鍵を持つ個人のみが復号することができる。公開鍵暗号化方式の代表的なものにはＲＳＡ（Rivest-Shamir-Adleman）暗号がある。このような暗号化方式を利用することにより、暗号化コンテンツを正規ユーザに対してのみ復号可能とするシステムが可能となる。

上記のようなコンテンツ配信システムでは、コンテンツを暗号化してユーザにネットワーク、あるいはＤＶＤ、ＣＤ等の記録媒体に格納して提供し、暗号化コンテンツを復号するコンテンツキーを正当なユーザにのみ提供する構成が多く採用されている。コンテンツキー自体の不正なコピー等を防ぐためのコンテンツキーを暗号化して正当なユーザに提供し、正当なユーザのみが有する復号キーを用いて暗号化コンテンツキーを復号してコンテンツキーを使用可能とする構成が提案されている。

正当なユーザであるか否かの判定は、一般には、例えばコンテンツの送信者であるコンテンツプロバイダとユーザデバイス間において、コンテンツ、あるいはコンテンツキーの配信前に認証処理を実行することによって行う。一般的な認証処理においては、相手の確認を行うとともに、その通信でのみ有効なセッションキーを生成して、認証が成立した場合に、生成したセッションキーを用いてデー

タ、例えばコンテンツあるいはコンテンツキーを暗号化して通信を行う。認証方式には、共通鍵暗号方式を用いた相互認証と、公開鍵方式を使用した認証方式があるが、共通鍵を使った認証においては、システムワイドで共通な鍵が必要になり、更新処理等の際に不便である。また、公開鍵方式においては、計算負荷が大きくまた必要なメモリ量も大きくなり、各デバイスにこのような処理手段を設けることは望ましい構成とはいえない。

発明の開示

本発明では、上述のようなデータの送信者、受信者間の相互認証処理に頼ることなく、正当なユーザに対してのみ、安全にデータを送信することを可能とするとともに、階層的鍵配信ツリーを配信デバイスに応じて再構築して配信キーブロックに含まれるデータ量を削減することにより、暗号化キーのデータ量を削減し、データ送信の負荷を軽減するとともに、各デバイスにおける暗号化キー取得のための処理の軽減を可能とした情報処理システム、情報処理方法、及び情報記録媒体、並びにプログラム提供媒体を提供することを目的とする。

本発明に係る情報処理システムは、１以上の選択されたデバイスにおいてのみ利用可能な暗号化メッセージデータを配信する情報処理システムである。個々のデバイスは、複数の異なるデバイスをリーフとした階層ツリー構造における各ノードに固有のノードキーと各デバイス固有のリーフキーの異なるキーセットをそれぞれ保有するとともに、デバイスに対して配信される暗号化メッセージデータについての復号処理をキーセットを使用して実行する暗号処理手段を有する。デバイスに提供される暗号化メッセージデータは、階層ツリー構造の１つのノードを頂点ノードとし、頂点ノードの下位に連結されるノード及びリーフによって構成されるグループ内のノードキーの少なくともいずれかを更新した更新ノードキーと、更新ノードキーをグループのノードキーあるいはリーフキーによって暗号化した暗号化キーデータを含む有効化キーブロック（ＥＫＢ）の復号処理によって得られる更新ノードキーによって暗号化されたデータ構成である。有効化キーブロック（ＥＫＢ）は、暗号化キーデータによって構成されるデータ部と、デー

タ部に格納される暗号化キーデータの階層ツリー構造における位置識別データとしてのタグ部とを含む構成である。

さらに、本発明の情報処理システムの一実施態様において、有効化キーブロック（ＥＫＢ）に含まれる暗号化キーデータは、階層ツリー構造を構成するノードキーを下位ノードキー又は下位リーフキーを用いて暗号化したデータである。タグ部に格納される位置識別データは、有効化キーブロック（ＥＫＢ）に格納された１以上の暗号化キーデータ各々のノード位置の下位の左右ノード又はリーフ位置の暗号化キーデータの有無を示すタグとして構成されている。

さらに、本発明の情報処理システムの一実施態様において、有効化キーブロック（ＥＫＢ）に含まれる暗号化キーデータは、有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段とした簡略化した２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーのノード又はリーフに対応するキーのみに基づいて構成されている。タグ部に格納される位置識別データは、有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示すデータを含む構成である。

さらに、本発明の情報処理システムの一実施態様において、有効化キーブロック（ＥＫＢ）に含まれる暗号化キーデータは、有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段とした簡略化した２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーのノード又はリーフに対応するキーのみに基づいて構成されている。タグ部に格納される位置識別データは、有効化キーブロック（ＥＫＢ）に格納された１以上の暗号化キーデータ各々のノード位置の下位の左右ノード又はリーフ位置の暗号化キーデータの有無を示すタグと、タグに対応する暗号化キーの格納の有無を示すデータを含む。

さらに、本発明の情報処理システムの一実施態様において、再構築階層ツリーは、共通要素を持つデバイスの部分集合ツリーとして定義されるエンティティの頂点ノードであるサブルートを選択して構成されるツリーである。

さらに、本発明の情報処理システムの一実施態様において、有効化キーブロック（ＥＫＢ）に含まれる暗号化キーデータは、有効化キーブロック（ＥＫＢ）を

復号可能な末端ノード又はリーフを最下段とした簡略化した多分岐型ツリーにおいて、末端ノード又はリーフと、多分岐型ツリーの頂点とを直接接続するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーの頂点ノード及び末端ノード又はリーフに対応するキーのみに基づいて構成されている。タグ部に格納される位置識別データは、有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示すデータを含む構成である。

さらに、本発明の情報処理システムの一実施態様において、再構築階層ツリーは、簡略化した多分岐型ツリーを構成する頂点ノードと、簡略化したツリーを構成する末端ノード又はリーフとを直接接続した３以上の分岐を持つツリーである。

さらに、本発明の情報処理システムの一実施態様において、デバイスにおける暗号処理手段は、有効化キーブロック（ＥＫＢ）のタグ部のデータにより、暗号化キーデータを順次抽出して、復号処理を実行し、更新ノードキーを取得し、取得した更新ノードキーにより暗号化メッセージデータの復号を実行する構成である。

さらに、本発明の情報処理システムの一実施態様において、メッセージデータは、コンテンツデータを復号するための復号鍵として使用可能なコンテンツキーである。

さらに、本発明の情報処理システムの一実施態様において、メッセージデータは、認証処理において用いられる認証キーである。

さらに、本発明の情報処理システムの一実施態様において、メッセージデータは、コンテンツのインテグリティ・チェック値（ＩＣＶ）生成キーである。

さらに、本発明の情報処理システムの一実施態様において、メッセージデータは、プログラムコードである。

本発明に係る情報処理方法は、１以上の選択されたデバイスにおいてのみ利用可能な暗号化メッセージデータを配信する情報処理方法である。この情報処理方法は、複数の異なるデバイスをリーフとした階層ツリー構造の１つのノードを頂点ノードとし、頂点ノードの下位に連結されるノード及びリーフによって構成されるグループ内のノードキーの少なくともいずれかを更新した更新ノードキーと、更新ノードキーをグループのノードキーあるいはリーフキーによって暗号化した

暗号化キーデータを含むデータ部と、データ部に格納される暗号化キーデータの階層ツリー構造における位置識別データとしてのタグ部とを含む有効化キーブロック（ＥＫＢ）を生成する有効化キーブロック（ＥＫＢ）生成ステップと、更新ノードキーによって暗号化したメッセージデータを生成してデバイスに対して配信するメッセージデータ配信ステップとを有する。

さらに、本発明の情報処理方法の一実施態様において、階層ツリー構造における各ノードに固有のノードキーと各デバイス固有のリーフキーの異なるキーセットをそれぞれ保有するデバイスにおいて、暗号化メッセージデータについての復号処理をキーセットを使用して実行する復号処理ステップを有する。

さらに、本発明の情報処理方法の一実施態様において、有効化キーブロック（ＥＫＢ）生成ステップは、階層ツリー構造を構成するノードキーを下位ノードキー又は下位リーフキーを用いて暗号化して暗号化キーデータを生成するステップと、有効化キーブロック（ＥＫＢ）に格納される１以上の暗号化キーデータ各々のノード位置の下位の左右位置のノード又はリーフ位置の暗号化キーデータの有無を示すタグを生成してタグ部に格納するステップとを含む。

さらに、本発明の情報処理方法の一実施態様において、有効化キーブロック（ＥＫＢ）生成ステップは、有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段とした簡略化した２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築階層ツリーを生成するステップと、再構築階層ツリーの構成ノード又はリーフに対応するキーのみに基づいて有効化キーブロック（ＥＫＢ）を生成するステップと、有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示すデータをタグ部に格納するステップとを含む。

さらに、本発明の情報処理方法の一実施態様において、再構築階層ツリーを生成するステップは、共通要素を持つデバイスの部分集合ツリーとして定義されるエンティティの頂点ノードであるサブルートを選択して実行されるツリー生成処理である。

さらに、本発明の情報処理方法の一実施態様において、有効化キーブロック（ＥＫＢ）生成ステップは、有効化キーブロック（ＥＫＢ）を復号可能な末端ノ

ード又はリーフを最下段とした簡略化した多分岐型ツリーにおいて、末端ノード又はリーフと、多分岐型ツリーの頂点とを直接接続するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーを生成するステップと、有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示すデータをタグ部に格納するステップとを含む。

さらに、本発明の情報処理方法の一実施態様において、再構築階層ツリーの生成ステップにおいて生成する再構築階層ツリーは、簡略化した多分岐型ツリーを構成する頂点ノードと、簡略化したツリーを構成する末端ノード又はリーフとを直接、接続した３以上の分岐を持つツリーとして生成する。

さらに、本発明の情報処理方法の一実施態様において、復号処理ステップは、有効化キーブロック（ＥＫＢ）のタグ部に格納された位置識別データに基づいてデータ部に格納された暗号化キーデータを順次抽出して順次復号処理を実行することにより更新ノードキーを取得する更新ノードキー取得ステップと、更新ノードキーにより暗号化メッセージデータの復号を実行するメッセージデータ復号ステップとを含む。

さらに、本発明の情報処理方法の一実施態様において、メッセージデータは、コンテンツデータを復号するための復号鍵として使用可能なコンテンツキーである。

さらに、本発明の情報処理方法の一実施態様において、メッセージデータは、認証処理において用いられる認証キーである。

さらに、本発明の情報処理方法の一実施態様において、メッセージデータは、コンテンツのインテグリティ・チェック値（ＩＣＶ）生成キーである。

さらに、本発明の情報処理方法の一実施態様において、メッセージデータは、プログラムコードである。

さらに、本発明に係る情報記録媒体は、データを格納した情報記録媒体である。この情報記録媒体は、複数の異なるデバイスをリーフとした階層ツリー構造の１つのノードを頂点ノードとし、頂点ノードの下位に連結されるノード及びリーフによって構成されるグループ内のノードキーの少なくともいずれかを更新した更新ノードキーと、更新ノードキーをグループのノードキーあるいはリーフキーに

よって暗号化した暗号化キーデータによって構成されるデータ部と、データ部に格納される暗号化キーデータの階層ツリー構造における位置識別データとしてのタグ部とを含む有効化キーブロック（ＥＫＢ）と、更新ノードキーによって暗号化したメッセージデータとを格納している。

さらに、本発明の情報記録媒体の一実施態様において、有効化キーブロック（ＥＫＢ）に含まれる暗号化キーデータは、階層ツリー構造を構成するノードキーを下位ノードキー又は下位リーフキーを用いて暗号化したデータである。タグ部に格納される位置識別データは、有効化キーブロック（ＥＫＢ）に格納された１以上の暗号化キーデータ各々のノード位置の下位の左右位置のノード又はリーフ位置の暗号化キーデータの有無を示すタグとして構成されている。

さらに、本発明の情報記録媒体の一実施態様において、有効化キーブロック（ＥＫＢ）に含まれる暗号化キーデータは、有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段とした簡略化した２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーのノード又はリーフに対応するキーのみに基づいて構成されている。タグ部に格納される位置識別データは、有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示すデータを含む構成である。

本発明に係るプログラム提供媒体は、複数の異なるデバイスをリーフとした階層ツリー構造の１つのノードを頂点ノードとし、頂点ノードの下位に連結されるノード及びリーフによって構成されるグループ内のノードキーの少なくともいずれかを更新した更新ノードキーと、更新ノードキーをグループのノードキーあるいはリーフキーによって暗号化した有効化キーブロック（ＥＫＢ）の生成処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを提供するプログラム提供媒体である。コンピュータ・プログラムは、有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段とした簡略化した２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築階層ツリーを生成するステップと、再構築階層ツリーの構成ノード又はリーフに対応するキーのみに基づいて有効化キーブロック（ＥＫＢ）を生成するステップと、有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示す

データをタグ部に格納するステップとを含む。

本発明では、ツリー（木）構造の階層的構造の暗号化鍵配信構成を用いることにより、キー更新に必要な配信メッセージ量を小さく抑えている。すなわち、各機器をｎ分木の各葉（リーフ）に配置した構成の鍵配信方法を用い、記録媒体若しくは通信回線を介して、例えばコンテンツデータの暗号鍵であるコンテンツキー若しくは認証処理に用いる認証キー、あるいはプログラムコード等を有効化キーブロックとともに配信する構成としている。

さらに、有効化キーブロックを暗号化キーデータ部と、暗号化キーの位置を示すタグ部とによって構成し、データ量を少なくし、デバイスにおける復号処理を用意かつ迅速に実行することを可能としている。本構成により、正当なデバイスのみが復号可能なデータを安全に配信することが可能となる。

なお、本発明に係るプログラム提供媒体は、例えば、様々なプログラム・コードを実行可能な汎用コンピュータ・システムに対して、コンピュータ・プログラムをコンピュータ可読な形式で提供する媒体である。媒体は、ＣＤやＦＤ、ＭＯなどの記録媒体、あるいは、ネットワークなどの伝送媒体など、その形態は特に限定されない。

このようなプログラム提供媒体は、コンピュータ・システム上で所定のコンピュータ・プログラムの機能を実現するための、コンピュータ・プログラムと提供媒体との構造上又は機能上の協働的関係を定義したものである。換言すれば、提供媒体を介してコンピュータ・プログラムをコンピュータ・システムにインストールすることによって、コンピュータ・システム上では協働的作用が発揮され、本発明の他の側面と同様の作用効果を得ることができるのである。

本発明のさらに他の目的、特徴や利点は、後述する本発明の実施例や添付する図面に基づくより詳細な説明によって明らかになるであろう。

図面の簡単な説明

図１は、本発明の情報処理システムの構成例を説明する図である。

図２は、本発明の情報処理システムにおいて適用可能な記録再生装置の構成例

を示すブロック図である。

図３は、本発明の情報処理システムにおける各種キー、データの暗号化処理について説明するツリー構成図である。

図４Ａ及び図４Ｂは、本発明の情報処理システムにおける各種キー、データの配布に使用される有効化キーブロック（ＥＫＢ）の例を示す図である。

図５は、本発明の情報処理システムにおけるコンテンツキーの有効化キーブロック（ＥＫＢ）を使用した配布例と復号処理例を示す図である。

図６は、本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）のフォーマット例を示す図である。

図７Ａ乃至図７Ｃは、本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）のタグの構成を説明する図である。

図８Ａ及び図８Ｂは、本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、コンテンツキー、コンテンツを併せて配信するデータ構成例を示す図である。

図９は、本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、コンテンツキー、コンテンツを併せて配信した場合のデバイスでの処理例を示す図である。

図１０は、本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）とコンテンツを記録媒体に格納した場合の対応について説明する図である。

図１１Ａ及び図１１Ｂは、本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、コンテンツキーを送付する処理を従来の送付処理と比較した図である。

図１２は、本発明の情報処理システムにおいて適用可能な共通鍵暗号方式による認証処理シーケンスを示す図である。

図１３は、本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、認証キーを併せて配信するデータ構成と、デバイスでの処理例を示す図（その１）である。

図１４は、本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、認証キーを併せて配信するデータ構成と、デバイスでの処理例を示す図（そ

の２）である。

図１５は、本発明の情報処理システムにおいて適用可能な公開鍵暗号方式による認証処理シーケンスを示す図である。

図１６は、本発明の情報処理システムにおいて公開鍵暗号方式による認証処理を用いて有効化キーブロック（ＥＫＢ）と、コンテンツキーを併せて配信する処理を示す図である。

図１７は、本発明の情報処理システムにおいて有効化キーブロック（ＥＫＢ）と、暗号化プログラムデータを併せて配信する処理を示す図である。

図１８は、本発明の情報処理システムにおいて適用可能なコンテンツ・インテグリティ・チェック値（ＩＣＶ）の生成に使用するＭＡＣ値生成例を示す図である。

図１９は、本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、ＩＣＶ生成キーを併せて配信するデータ構成と、デバイスでの処理例を示す図（その１）である。

図２０は、本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、ＩＣＶ生成キーを併せて配信するデータ構成と、デバイスでの処理例を示す図（その２）である。

図２１Ａ及び図２１Ｂは、本発明の情報処理システムにおいて適用可能なコンテンツ・インテグリティ・チェック値（ＩＣＶ）をメディアに格納した場合のコピー防止機能を説明する図である。

図２２は、本発明の情報処理システムにおいて適用可能なコンテンツ・インテグリティ・チェック値（ＩＣＶ）をコンテンツ格納媒体と別に管理する構成を説明する図である。

図２３は、本発明の情報処理システムにおける階層ツリー構造のカテゴリ分類の例を説明する図である。

図２４Ａ及び図２４Ｂは、本発明の情報処理システムにおける簡略化有効化キーブロック（ＥＫＢ）の生成過程を説明する図である。

図２５Ａ及び図２５Ｂは、本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）の生成過程を説明する図である。

図２６Ａ及び図２６Ｂは、本発明の情報処理システムにおける簡略化有効化キーブロック（ＥＫＢ）（例１）を説明する図である。

図２７Ａ及び図２７Ｂは、本発明の情報処理システムにおける簡略化有効化キーブロック（ＥＫＢ）（例２）を説明する図である。

図２８Ａ乃至図２８Ｃは、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成について説明する図である。

図２９Ａ乃至図２９Ｃは、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成の詳細について説明する図である。

図３０Ａ及び図３０Ｂは、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成について説明する図である。

図３１は、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのリザーブノードについて説明する図である。

図３２は、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成での新規エンティティ登録処理シーケンスについて説明する図である。

図３３は、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成での新規エンティティと上位エンティティの関係について説明する図である。

図３４Ａ及び図３４Ｂは、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成で用いるサブＥＫＢについて説明する図である。

図３５Ａ乃至図３５Ｄは、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのデバイスリボーク処理について説明する図である。

図３６は、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのデバイスリボーク処理シーケンスについて説明する図である。

図３７Ａ及び図３７Ｂは、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのデバイスリボーク時の更新サブＥＫＢについて説明する図である。

図３８Ａ乃至図３８Ｄは、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのエンティティリボーク処理について説明する図である。

図３９は、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのエンティティリボーク処理シーケンスについて説明する図である。

図４０は、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのリボークエンティティと上位エンティティの関係について説明する図である。

図４１は、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのケイパビリティ設定について説明する図である。

図４２は、本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのケイパビリティ設定について説明する図である。

図４３Ａ及び図４３Ｂは、本発明の情報処理システムにおけるキー発行センタ（ＫＤＣ）の管理するケイパビリティ管理テーブル構成を説明する図である。

図４４は、本発明の情報処理システムにおけるキー発行センタ（ＫＤＣ）の管理するケイパビリティ管理テーブルに基づくＥＫＢ生成処理フロー図である。

図４５は、本発明の情報処理システムにおける新規エンティティ登録時のケイパビリティ通知処理を説明する図である。

発明を実施するための最良の形態

［システム概要］

図１に本発明のデータ処理システムが適用可能なコンテンツ配信システム例を示す。コンテンツの配信側１０は、コンテンツ受信側２０の有する様々なコンテンツ再生可能な機器に対してコンテンツ、あるいはコンテンツキーを暗号化して送信する。受信側２０における機器では、受信した暗号化コンテンツ、あるいは暗号化コンテンツキー等を復号してコンテンツあるいはコンテンツキーを取得して、画像データ、音声データの再生、あるいは各種プログラムの実行等を行う。コンテンツの配信側１０とコンテンツ受信側２０との間のデータ交換は、インターネット等のネットワークを介して、あるいはＤＶＤ、ＣＤ等の流通可能な記憶媒体を介して実行される。

コンテンツの配信側１０は例えば所謂サーバとして構成され、ハードディスク

ドライブなどの情報記憶手段やＣＰＵなどの情報処理装置を有する既存のパーソナルコンピュータで構成される。コンテンツの配信側１０は後述するコンテンツプロバイダとして、あるいはサービスプロバイダやアプリケーションプロバイダが同等の機能を有していてもよい。コンテンツの配信側１０のデータ配信手段としては、インターネット１１、衛星放送１２、電話回線１３、ＤＶＤ、ＣＤ等のメディア１４等があり、一方、コンテンツ受信側２０のデバイスとしては、パーソナルコンピュータ（ＰＣ）２１、ポータブルデバイス（ＰＤ）２２、携帯電話、ＰＤＡ（Personal Digital Assistants）等の携帯機器２３、ＤＶＤ、ＣＤプレーヤ等の記録再生器２４、ゲーム端末等の再生専用器２５等がある。これらコンテンツ受信側２０の各デバイスは、コンテンツ配信側１０から提供されたコンテンツをネットワーク等の通信手段あるいは、あるいはメディア３０から取得する。

［デバイス構成］

図２に、図１に示すコンテンツ受信側２０のデバイスの一例として、記録再生装置１００の構成ブロック図を示す。記録再生装置１００は、入出力Ｉ／Ｆ（Interface）１２０、ＭＰＥＧ（Moving Picture Experts Group）コーデック１３０、Ａ／Ｄ，Ｄ／Ａコンバータ１４１を備えた入出力Ｉ／Ｆ（Interface）１４０、暗号処理手段１５０、ＲＯＭ（Read Only Memory）１６０、ＣＰＵ（Central Processing Unit）１７０、メモリ１８０、記録媒体１９５のドライブ１９０を有し、これらはバス１１０によって相互に接続されている。

入出力Ｉ／Ｆ１２０は、外部から供給される画像、音声、プログラム等の各種コンテンツを構成するディジタル信号を受信し、バス１１０上に出力するとともに、バス１１０上のディジタル信号を受信し、外部に出力する。ＭＰＥＧコーデック１３０は、バス１１０を介して供給されるＭＰＥＧ符号化されたデータを、ＭＰＥＧデコードし、入出力Ｉ／Ｆ１４０に出力するとともに、入出力Ｉ／Ｆ１４０から供給されるディジタル信号をＭＰＥＧエンコードしてバス１１０上に出力する。入出力Ｉ／Ｆ１４０は、Ａ／Ｄ，Ｄ／Ａコンバータ１４１を内蔵している。入出力Ｉ／Ｆ１４０は、外部から供給されるコンテンツとしてのアナログ信号を受信し、Ａ／Ｄ，Ｄ／Ａコンバータ１４１でＡ／Ｄ（Analog Digital）変換することで、ディジタル信号として、ＭＰＥＧコーデック１３０に出力するとと

もに、ＭＰＥＧコーデック１３０からのディジタル信号を、Ａ／Ｄ，Ｄ／Ａコンバータ１４１でＤ／Ａ（Digital Analog）変換することで、アナログ信号として、外部に出力する。

暗号処理手段１５０は、例えば、１チップのＬＳＩ（Large Scale Integrated Curcuit）で構成され、バス１１０を介して供給されるコンテンツとしてのディジタル信号の暗号化、復号処理、あるいは認証処理を実行し、暗号データ、復号データ等をバス１１０上に出力する構成を持つ。なお、暗号処理手段１５０は１チップＬＳＩに限らず、各種のソフトウェア又はハードウェアを組み合わせた構成によって実現することも可能である。ソフトウェア構成による処理手段としての構成については後段で説明する。

ＲＯＭ１６０は、記録再生装置によって処理されるプログラムデータを格納する。ＣＰＵ１７０は、ＲＯＭ１６０、メモリ１８０に記憶されたプログラムを実行することで、ＭＰＥＧコーデック１３０や暗号処理手段１５０等を制御する。メモリ１８０は、例えば、不揮発性メモリで、ＣＰＵ１７０が実行するプログラムや、ＣＰＵ１７０の動作上必要なデータ、さらにデバイスによって実行される暗号処理に使用されるキーセットを記憶する。キーセットについては後段で説明する。ドライブ１９０は、ディジタルデータを記録再生可能な記録媒体１９５を駆動することにより、記録媒体１９５からディジタルデータを読み出し（再生し）、バス１１０上に出力するとともに、バス１１０を介して供給されるディジタルデータを、記録媒体１９５に供給して記録させる。

記録媒体１９５は、例えば、ＤＶＤ、ＣＤ等の光ディスク、光磁気ディスク、磁気ディスク、磁気テープ、あるいはＲＡＭ等の半導体メモリ等のディジタルデータの記憶可能な媒体であり、本実施の形態では、ドライブ１９０に対して着脱可能な構成であるとする。ただし、記録媒体１９５は、記録再生装置１００に内蔵する構成としてもよい。

なお、図２に示す暗号処理手段１５０は、１つのワンチップＬＳＩとして構成してもよく、また、ソフトウェア、ハードウェアを組み合わせた構成によって実現する構成としてもよい。

［キー配信構成としてのツリー（木）構造について］

次に、図１に示すコンテンツ配信側１０からコンテンツ受信側２０の各デバイスに暗号データを配信する場合における各デバイスにおける暗号処理鍵の保有構成及びデータ配信構成を図３を用いて説明する。

図３の最下段に示すナンバ０～１５がコンテンツ受信側２０の個々のデバイスである。すなわち図３に示す階層ツリー（木）構造の各葉（リーフ：leaf）がそれぞれのデバイスに相当する。

各デバイス０～１５は、製造時あるいは出荷時、あるいはその後において、図３に示す階層ツリー（木）構造における、自分のリーフからルートに至るまでのノードに割り当てられた鍵（ノードキー）及び各リーフのリーフキーからなるキーセットをメモリに格納する。図３の最下段に示すＫ００００～Ｋ１１１１が各デバイス０～１５にそれぞれ割り当てられたリーフキーであり、最上段のＫＲ（ルートキー）から、最下段から２番目の節（ノード）に記載されたキー：ＫＲ～Ｋ１１１をノードキーとする。

図３に示すツリー構成において、例えばデバイス０はリーフキーＫ００００と、ノードキー：Ｋ０００、Ｋ００、Ｋ０、ＫＲを所有する。デバイス５はＫ０１０１、Ｋ０１０、Ｋ０１、Ｋ０、ＫＲを所有する。デバイス１５は、Ｋ１１１１、Ｋ１１１、Ｋ１１、Ｋ１、ＫＲを所有する。なお、図３のツリーにはデバイスが０～１５の１６個のみ記載され、ツリー構造も４段構成の均衡のとれた左右対称構成として示しているが、さらに多くのデバイスがツリー中に構成され、また、ツリーの各部において異なる段数構成を持つことが可能である。

また、図３のツリー構造に含まれる各デバイスには、様々な記録媒体、例えば、デバイス埋め込み型あるいはデバイスに着脱自在に構成されたＤＶＤ、ＣＤ、ＭＤ、フラッシュメモリ等を使用する様々なタイプのデバイスが含まれている。さらに、様々なアプリケーションサービスが共存可能である。このような異なるデバイス、異なるアプリケーションの共存構成の上に図３に示すコンテンツあるいは鍵配布構成である階層ツリー構造が適用される。

これらの様々なデバイス、アプリケーションが共存するシステムにおいて、例えば図３の点線で囲んだ部分、すなわちデバイス０，１，２，３を同一の記録媒体を用いる１つのグループとして設定する。例えば、この点線で囲んだグループ

内に含まれるデバイスに対しては、まとめて、共通のコンテンツを暗号化してプロバイダから送付したり、各デバイス共通に使用するコンテンツキーを送付したり、あるいは各デバイスからプロバイダあるいは決済機関等にコンテンツ料金の支払データをやはり暗号化して出力するといった処理が実行される。コンテンツプロバイダ、あるいは決済処理機関等、各デバイスとのデータ送受信を行う機関は、図３の点線で囲んだ部分、すなわちデバイス０，１，２，３を１つのグループとして一括してデータを送付する処理を実行する。このようなグループは、図３のツリー中に複数存在する。コンテンツプロバイダ、あるいは決済処理機関等、各デバイスとのデータ送受信を行う機関は、メッセージデータ配信手段として機能する。

なお、ノードキー、リーフキーは、ある１つの鍵管理センタによって統括して管理してもよいし、各グループに対する様々なデータ送受信を行うプロバイダ、決済機関等のメッセージデータ配信手段によってグループ毎に管理する構成としてもよい。これらのノードキー、リーフキーは例えばキーの漏洩等の場合に更新処理が実行され、この更新処理は鍵管理センタ、プロバイダ、決済機関等が実行する。

このツリー構造において、図３から明らかなように、１つのグループに含まれる３つのデバイス０，１，２，３はノードキーとして共通のキーＫ００、Ｋ０、ＫＲを保有する。このノードキー共有構成を利用することにより、例えば共通のコンテンツキーをデバイス０，１，２，３のみに提供することが可能となる。例えば、共通に保有するノードキーＫ００自体をコンテンツキーとして設定すれば、新たな鍵送付を実行することなくデバイス０，１，２，３のみが共通のコンテンツキーの設定が可能である。また、新たなコンテンツキーＫｃｏｎをノードキーＫ００で暗号化した値Ｅｎｃ（Ｋ００，Ｋｃｏｎ）を、ネットワークを介してあるいは記録媒体に格納してデバイス０，１，２，３に配布すれば、デバイス０，１，２，３のみが、それぞれのデバイスにおいて保有する共有ノードキーＫ００を用いて暗号Ｅｎｃ（Ｋ００，Ｋｃｏｎ）を解いてコンテンツキー：Ｋｃｏｎを得ることが可能となる。なお、Ｅｎｃ（Ｋａ，Ｋｂ）はＫｂをＫａによって暗号化したデータであることを示す。

また、ある時点ｔにおいて、デバイス３の所有する鍵：Ｋ００１１，Ｋ００１，Ｋ００，Ｋ０，ＫＲが攻撃者（ハッカー）により解析されて露呈したことが発覚した場合、それ以降、システム（デバイス０，１，２，３のグループ）で送受信されるデータを守るために、デバイス３をシステムから切り離す必要がある。そのためには、ノードキー：Ｋ００１，Ｋ００，Ｋ０，ＫＲをそれぞれ新たな鍵Ｋ（ｔ）００１，Ｋ（ｔ）００，Ｋ（ｔ）０，Ｋ（ｔ）Ｒに更新し、デバイス０，１，２にその更新キーを伝える必要がある。ここで、Ｋ（ｔ）ａａａは、鍵Ｋａａａの世代（Generation）：ｔの更新キーであることを示す。

更新キーの配布処理ついて説明する。キーの更新は、例えば、図４Ａに示す有効化キーブロック（ＥＫＢ：Enabling Key Block）と呼ばれるブロックデータによって構成されるテーブルを例えばネットワーク、あるいは記録媒体に格納してデバイス０，１，２に供給することによって実行される。なお、有効化キーブロック（ＥＫＢ）は、図３に示すようなツリー構造を構成する各リーフに対応するデバイスに新たに更新されたキーを配布するための暗号化キーによって構成される。有効化キーブロック（ＥＫＢ）は、キー更新ブロック（ＫＲＢ：Key Renewal Block）と呼ばれることもある。

図４Ａに示す有効化キーブロック（ＥＫＢ）には、ノードキーの更新の必要なデバイスのみが更新可能なデータ構成を持つブロックデータとして構成される。図４Ａ及び図４Ｂの例は、図３に示すツリー構造中のデバイス０，１，２において、世代ｔの更新ノードキーを配布することを目的として形成されたブロックデータである。図３から明らかなように、デバイス０，デバイス１は、更新ノードキーとしてＫ（ｔ）００、Ｋ（ｔ）０、Ｋ（ｔ）Ｒが必要であり、デバイス２は、更新ノードキーとしてＫ（ｔ）００１、Ｋ（ｔ）００、Ｋ（ｔ）０、Ｋ（ｔ）Ｒが必要である。

図４ＡのＥＫＢに示されるようにＥＫＢには複数の暗号化キーが含まれる。最下段の暗号化キーは、Ｅｎｃ（Ｋ００１０，Ｋ（ｔ）００１）である。これはデバイス２の持つリーフキーＫ００１０によって暗号化された更新ノードキーＫ（ｔ）００１であり、デバイス２は、自身の持つリーフキーによってこの暗号化キーを復号し、Ｋ（ｔ）００１を得ることができる。また、復号により得たＫ

（ｔ）００１を用いて、図４Ａの下から２段目の暗号化キーＥｎｃ（Ｋ（ｔ）００１，Ｋ（ｔ）００）を復号可能となり、更新ノードキーＫ（ｔ）００を得ることができる。以下順次、図４Ａの上から２段目の暗号化キーＥｎｃ（Ｋ（ｔ）００，Ｋ（ｔ）０）を復号し、更新ノードキーＫ（ｔ）０、図４Ａの上から１段目の暗号化キーＥｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）Ｒ）を復号しＫ（ｔ）Ｒを得る。一方、デバイスＫ００００．Ｋ０００１は、ノードキーＫ０００は更新する対象に含まれておらず、更新ノードキーとして必要なのは、Ｋ（ｔ）００、Ｋ（ｔ）０、Ｋ（ｔ）Ｒである。デバイスＫ００００．Ｋ０００１は、図４Ａの上から３段目の暗号化キーＥｎｃ（Ｋ０００，Ｋ（ｔ）００）を復号しＫ（ｔ）００を取得し、以下、図４Ａの上から２段目の暗号化キーＥｎｃ（Ｋ（ｔ）００，Ｋ（ｔ）０）を復号し、更新ノードキーＫ（ｔ）０、図４Ａの上から１段目の暗号化キーＥｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）Ｒ）を復号しＫ（ｔ）Ｒを得る。このようにして、デバイス０，１，２は更新した鍵Ｋ（ｔ）Ｒを得ることができる。なお、図４Ａのインデックスは、復号キーとして使用するノードキー、リーフキーの絶対番地を示す。

図３に示すツリー構造の上位段のノードキー：Ｋ（ｔ）０，Ｋ（ｔ）Ｒの更新が不要であり、ノードキーＫ００のみの更新処理が必要である場合には、図４Ｂの有効化キーブロック（ＥＫＢ）を用いることで、更新ノードキーＫ（ｔ）００をデバイス０，１，２に配布することができる。

図４Ｂに示すＥＫＢは、例えば特定のグループにおいて共有する新たなコンテンツキーを配布する場合に利用可能である。具体例として、図３に点線で示すグループ内のデバイス０，１，２，３がある記録媒体を用いており、新たな共通のコンテンツキーＫ（ｔ）ｃｏｎが必要であるとする。このとき、デバイス０，１，２，３の共通のノードキーＫ００を更新したＫ（ｔ）００を用いて新たな共通の更新コンテンツキー：Ｋ（ｔ）ｃｏｎを暗号化したデータＥｎｃ（Ｋ（ｔ）００，Ｋ（ｔ）ｃｏｎ）を図４Ｂに示すＥＫＢとともに配布する。この配布により、デバイス４など、その他のグループの機器においては復号されないデータとしての配布が可能となる。

すなわち、デバイス０，１，２はＥＫＢを処理して得たＫ（ｔ）００を用いて

上記暗号文を復号すれば、ｔ時点でのコンテンツキーＫ（ｔ）ｃｏｎを得ることが可能になる。

［ＥＫＢを使用したコンテンツキーの配布］

図５に、ｔ時点でのコンテンツキーＫ（ｔ）ｃｏｎを得る処理例として、Ｋ（ｔ）００を用いて新たな共通のコンテンツキーＫ（ｔ）ｃｏｎを暗号化したデータＥｎｃ（Ｋ（ｔ）００，Ｋ（ｔ）ｃｏｎ）と図４Ｂに示すＥＫＢとを記録媒体を介して受領したデバイス０の処理を示す。すなわちＥＫＢによる暗号化メッセージデータをコンテンツキーＫ（ｔ）ｃｏｎとした例である。

図５に示すように、デバイス０は、記録媒体に格納されている世代：ｔ時点のＥＫＢと自分が予め格納しているノードキーＫ０００を用いて上述したと同様のＥＫＢ処理により、ノードキーＫ（ｔ）００を生成する。さらに、復号した更新ノードキーＫ（ｔ）００を用いて更新コンテンツキーＫ（ｔ）ｃｏｎを復号して、後にそれを使用するために自分だけが持つリーフキーＫ０００００で暗号化して格納する。

［ＥＫＢのフォーマット］

図６に有効化キーブロック（ＥＫＢ）のフォーマット例を示す。バージョン６０１は、有効化キーブロック（ＥＫＢ）のバージョンを示す識別子である。なお、バージョンは最新のＥＫＢを識別する機能とコンテンツとの対応関係を示す機能を持つ。デプスは、有効化キーブロック（ＥＫＢ）の配布先のデバイスに対する階層ツリーの階層数を示す。データポインタ６０３は、有効化キーブロック（ＥＫＢ）中のデータ部の位置を示すポインタであり、タグポインタ６０４はタグ部の位置、署名ポインタ６０５は署名の位置を示すポインタである。

データ部６０６は、例えば更新するノードキーを暗号化したデータを格納する。例えば図５に示すような更新されたノードキーに関する各暗号化キー等を格納する。

タグ部６０７は、データ部に格納された暗号化されたノードキー、リーフキーの位置関係を示すタグである。このタグの付与ルールを図７Ａ乃至図７Ｃを用いて説明する。図７Ａ乃至図７Ｃでは、データとして先に図４Ａで説明した有効化キーブロック（ＥＫＢ）を送付する例を示している。この時のデータは、図７Ｂ

に示すようになる。このときの暗号化キーに含まれるトップノードのアドレスをトップノードアドレスとする。この場合は、ルートキーの更新キーＫ（ｔ）Ｒが含まれているので、トップノードアドレスはＫＲとなる。このとき、例えば最上段のデータＥｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）Ｒ）は、図７Ａに示す階層ツリーに示す位置にある。ここで、次のデータは、Ｅｎｃ（Ｋ（ｔ）００，Ｋ（ｔ）０）であり、ツリー上では前のデータの左下の位置にある。データがある場合は、タグが０、ない場合は１が設定される。タグは｛左（Ｌ）タグ，右（Ｒ）タグ｝として設定される。最上段のデータＥｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）Ｒ）の左にはデータがあるので、Ｌタグ＝０、右にはデータがないので、Ｒタグ＝１となる。以下、全てのデータにタグが設定され、図７Ｃに示すデータ列、及びタグ列が構成される。

タグは、データＥｎｃ（Ｋｘｘｘ，Ｋｙｙｙ）がツリー構造のどこに位置しているのかを示すために設定されるものである。データ部に格納されるキーデータＥｎｃ（Ｋｘｘｘ，Ｋｙｙｙ）．．．は、単純に暗号化されたキーの羅列データに過ぎないので、上述したタグによってデータとして格納された暗号化キーのツリー上の位置を判別可能としたものである。上述したタグを用いずに、先の図４Ａ及び図４Ｂで説明した構成のように暗号化データに対応させたノード・インデックスを用いて、例えば、

０：Ｅｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）ｒｏｏｔ）

００：Ｅｎｃ（Ｋ（ｔ）００，Ｋ（ｔ）０）

０００：Ｅｎｃ（Ｋ（（ｔ）０００，Ｋ（ｔ）００））

．．．のようなデータ構成とすることも可能であるが、このようなインデックスを用いた構成とすると冗長なデータとなりデータ量が増大し、ネットワークを介する配信等においては好ましくない。これに対し、上述したタグをキー位置を示す索引データとして用いることにより、少ないデータ量でキー位置の判別が可能となる。

図６に戻って、ＥＫＢフォーマットについてさらに説明する。署名（Signature）は、有効化キーブロック（ＥＫＢ）を発行した例えば鍵管理センタ、コンテンツプロバイダ、決済機関等が実行する電子署名である。ＥＫＢを受領したデバイ

スは署名検証によって正当な有効化キーブロック（ＥＫＢ）発行者が発行した有効化キーブロック（ＥＫＢ）であることを確認する。

［ＥＫＢを使用したコンテンツキー及びコンテンツの配信］

上述の例では、コンテンツキーのみをＥＫＢとともに送付する例について説明したが、コンテンツキーで暗号化したコンテンツと、コンテンツキー暗号キーで暗号化したコンテンツキーと、ＥＫＢによって暗号化したコンテンツキー暗号鍵を併せて送付する構成について以下説明する。

図８Ａ及び図８Ｂにこのデータ構成を示す。図８Ａに示す構成において、Ｅｎｃ（Ｋｃｏｎ，ｃｏｎｔｅｎｔ）８０１は、コンテンツ（Content）をコンテンツキー（Ｋｃｏｎ）で暗号化したデータであり、Ｅｎｃ（ＫＥＫ，Ｋｃｏｎ）８０２は、コンテンツキー（Ｋｃｏｎ）をコンテンツキー暗号キー（ＫＥＫ：Key Encryption Key）で暗号化したデータであり、Ｅｎｃ（ＥＫＢ，ＫＥＫ）８０３は、コンテンツキー暗号キーＫＥＫを有効化キーブロック（ＥＫＢ）によって暗号化したデータであることを示す。

ここで、コンテンツキー暗号キーＫＥＫは、図３で示すノードキー（Ｋ０００，Ｋ００…）、あるいはルートキー（ＫＲ）自体であってもよく、またノードキー（Ｋ０００，Ｋ００…）、あるいはルートキー（ＫＲ）によって暗号化されたキーであってもよい。

図８Ｂは、複数のコンテンツがメディアに記録され、それぞれが同じＥｎｃ（ＥＫＢ，ＫＥＫ）８０５を利用している場合の構成例を示す、このような構成においては、各データに同じＥｎｃ（ＥＫＢ，ＫＥＫ）を付加することなく、Ｅｎｃ（ＥＫＢ，ＫＥＫ）にリンクするリンク先を示すデータを各データに付加する構成とすることができる。

図９にコンテンツキー暗号キーＫＥＫを、図３に示すノードキーＫ００を更新した更新ノードキーＫ（ｔ）００として構成した場合の例を示す。この場合、図３の点線枠で囲んだグループにおいてデバイス３が、例えば鍵の漏洩によりリボーク（排除）されているとして、他のグループのメンバ、すなわち、デバイス０，１，２に対して図９に示す有効化キーブロック（ＥＫＢ）と、コンテンツキー（Ｋｃｏｎ）をコンテンツキー暗号キー（ＫＥＫ＝Ｋ（ｔ）００）で暗号化した

データと、コンテンツ（content）をコンテンツキー（Ｋｃｏｎ）で暗号化したデータとを配信することにより、デバイス０，１，２はコンテンツを得ることができる。

図９の右側には、デバイス０における復号手順を示してある。デバイス０は、まず、受領した有効化キーブロックから自身の保有するリーフキーＫ０００を用いた復号処理により、コンテンツキー暗号キー（ＫＥＫ＝Ｋ（ｔ）００）を取得する。次に、Ｋ（ｔ）００による復号によりコンテンツキーＫｃｏｎを取得し、さらにコンテンツキーＫｃｏｎによりコンテンツの復号を行う。これらの処理により、デバイス０はコンテンツを利用可能となる。デバイス１，２においても各々異なる処理手順でＥＫＢを処理することにより、コンテンツキー暗号キー（ＫＥＫ＝Ｋ（ｔ）００）を取得することが可能となり、同様にコンテンツを利用することが可能となる。

図３に示す他のグループのデバイス４，５，６…は、この同様のデータ（ＥＫＢ）を受信したとしても、自身の保有するリーフキー、ノードキーを用いてコンテンツキー暗号キー（ＫＥＫ＝Ｋ（ｔ）００）を取得することができない。同様にリボークされたデバイス３においても、自身の保有するリーフキー、ノードキーでは、コンテンツキー暗号キー（ＫＥＫ＝Ｋ（ｔ）００）を取得することができず、正当な権利を有するデバイスのみがコンテンツを復号して利用することが可能となる。

このように、ＥＫＢを利用したコンテンツキーの配送を用いれば、データ量を少なくして、かつ安全に正当権利者のみが復号可能とした暗号化コンテンツを配信することが可能となる。

なお、有効化キーブロック（ＥＫＢ）、コンテンツキー、暗号化コンテンツ等は、ネットワークを介して安全に配信することが可能な構成であるが、有効化キーブロック（ＥＫＢ）、コンテンツキー、暗号化コンテンツをＤＶＤ、ＣＤ等の記録媒体に格納してユーザに提供することも可能である。この場合、記録媒体に格納された暗号化コンテンツの復号には、同一の記録媒体に格納された有効化キーブロック（ＥＫＢ）の復号により得られるコンテンツキーを使用するように構成すれば、予め正当権利者のみが保有するリーフキー、ノードキーによってのみ

利用可能な暗号化コンテンツの配布処理、すなわち利用可能なユーザデバイスを限定したコンテンツ配布が簡易な構成で実現可能となる。

図１０に記録媒体に暗号化コンテンツとともに有効化キーブロック（ＥＫＢ）を格納した構成例を示す。図１０に示す例においては、記録媒体にコンテンツＣ１～Ｃ４が格納され、さらに各格納コンテンツに対応するの有効化キーブロック（ＥＫＢ）を対応付けたデータが格納され、さらにバージョンＭの有効化キーブロック（ＥＫＢ＿Ｍ）が格納されている。例えばＥＫＢ＿１はコンテンツＣ１を暗号化したコンテンツキーＫｃｏｎ１を生成するのに使用され、例えばＥＫＢ＿２はコンテンツＣ２を暗号化したコンテンツキーＫｃｏｎ２を生成するのに使用される。この例では、バージョンＭの有効化キーブロック（ＥＫＢ＿Ｍ）が記録媒体に格納されており、コンテンツＣ３，Ｃ４は有効化キーブロック（ＥＫＢ＿Ｍ）に対応付けられているので、有効化キーブロック（ＥＫＢ＿Ｍ）の復号によりコンテンツＣ３，Ｃ４のコンテンツキーを取得することができる。ＥＫＢ＿１、ＥＫＢ＿２はディスクに格納されていないので、新たな提供手段、例えばネットワーク配信、あるいは記録媒体による配信によってそれぞれのコンテンツキーを復号するために必要なＥＫＢ＿１，ＥＫＢ＿２を取得することが必要となる。

図１１Ａ及び図１１Ｂに、複数のデバイス間でコンテンツキーが流通する場合のＥＫＢを利用したコンテンツキーの配信と、従来のコンテンツキー配信処理の比較例を示す。図１１Ａが従来構成であり、図１１Ｂが本発明の有効化キーブロック（ＥＫＢ）を利用した例である。なお、図１１Ａ及び図１１ＢにおいてＫａ（Ｋｂ）は、ＫｂをＫａで暗号化したデータであることを示す。

図１１Ａに示すように、従来は、データ送受信者の正当性を確認し、またデータ送信の暗号化処理に使用するセッションキーＫｓｅｓを共有するために各デバイス間において、認証処理及び鍵交換処理（ＡＫＥ：Authentication and Key Exchange）を実行し、認証が成立したことを条件としてセッションキーＫｓｅｓでコンテンツキーＫｃｏｎを暗号化して送信する処理を行なっていた。

例えば図１１ＡのＰＣにおいては、受信したセッションキーで暗号化したコンテンツキーＫｓｅｓ（Ｋｃｏｎ）をセッションキーで復号してＫｃｏｎを得ることが可能であり、さらに取得したＫｃｏｎをＰＣ自体の保有する保存キーＫｓｔ

ｒで暗号化して自身のメモリに保存することが可能となる。

図１１Ａにおいて、コンテンツプロバイダは、図１１Ａの記録デバイス１１０１にのみデータを利用可能な形で配信したい場合でも、間にＰＣ、再生装置が存在する場合は、図１１Ａに示すように認証処理を実行し、それぞれのセッションキーでコンテンツキーを暗号化して配信するといった処理が必要となる。また、間に介在するＰＣ、再生装置においても認証処理において生成し共有することになったセッションキーを用いることで暗号化コンテンツキーを復号してコンテンツキーを取得可能となる。

一方、図１１Ｂの下段に示す有効化キーブロック（ＥＫＢ）を利用した例においては、コンテンツプロバイダから有効化キーブロック（ＥＫＢ）と、有効化キーブロック（ＥＫＢ）の処理によって得られるノードキー、又はルートキーによってコンテンツキーＫｃｏｎを暗号化したデータ（図の例ではＫｒｏｏｔ（Ｋｃｏｎ））を配信することにより、配信したＥＫＢの処理が可能な機器においてのみコンテンツキーＫｃｏｎを復号して取得することが可能になる。

従って、例えば図１１Ｂの右端にのみ利用可能な有効化キーブロック（ＥＫＢ）を生成して、その有効化キーブロック（ＥＫＢ）と、そのＥＫＢ処理によって得られるノードキー、又はルートキーによってコンテンツキーＫｃｏｎを暗号化したデータを併せて送ることにより、間に存在するＰＣ、再生機器等は、自身の有するリーフキー、ノードキーによっては、ＥＫＢの処理を実行することができない。従って、データ送受信デバイス間での認証処理、セッションキーの生成、セッションキーによるコンテンツキーＫｃｏｎの暗号化処理といった処理を実行することなく、安全に正当なデバイスに対してのみ利用可能なコンテンツキーを配信することが可能となる。

ＰＣ、記録再生器にも利用可能なコンテンツキーを配信したい場合は、それぞれにおいて処理可能な有効化キーブロック（ＥＫＢ）を生成して、配信することにより、共通のコンテンツキーを取得することが可能となる。

［有効化キーブロック（ＥＫＢ）を使用した認証キーの配信（共通鍵方式）］

上述の有効化キーブロック（ＥＫＢ）を使用したデータあるいはキーの配信に

おいて、デバイス間で転送される有効化キーブロック（ＥＫＢ）及びコンテンツあるいはコンテンツキーは常に同じ暗号化形態を維持しているため、データ伝走路を盗み出して記録し、再度、後で転送する、いわゆるリプレイアタックにより、不正コピーが生成される可能性がある。これを防ぐ構成としては、データ転送デバイス間において、従来と同様の認証処理及び鍵交換処理を実行することが有効な手段である。ここでは、この認証処理及び鍵交換処理を実行する際に使用する認証キーＫａｋｅを上述の有効化キーブロック（ＥＫＢ）を使用してデバイスに配信することにより、安全な秘密鍵として共有する認証キーを持ち、共通鍵方式に従った認証処理を実行する構成について説明する。すなわちＥＫＢによる暗号化メッセージデータを認証キーとした例である。

図１２に、共通鍵暗号方式を用いた相互認証方法（ISO/IEC 9798-2）を示す。図１２においては、共通鍵暗号方式としてＤＥＳを用いているが、共通鍵暗号方式であれば他の方式も可能である。図１２において、まず、Ｂが６４ビットの乱数Ｒｂを生成し、Ｒｂ及び自己のＩＤであるＩＤ（ｂ）をＡに送信する。これを受信したＡは、新たに６４ビットの乱数Ｒａを生成し、Ｒａ、Ｒｂ、ＩＤ（ｂ）の順に、ＤＥＳのＣＢＣモードで鍵Ｋａｂを用いてデータを暗号化し、Ｂに返送する。なお、鍵Ｋａｂは、Ａ及びＢに共通の秘密鍵としてそれぞれの記録素子内に格納する鍵である。ＤＥＳのＣＢＣモードを用いた鍵Ｋａｂによる暗号化処理は、例えばＤＥＳを用いた処理においては、初期値とＲａとの排他的論理和を求め、ＤＥＳ暗号化部において、鍵Ｋａｂを用いて暗号化し、暗号文Ｅ１を生成し、続けて暗号文Ｅ１とＲｂとの排他的論理和を求め、ＤＥＳ暗号化部において、鍵Ｋａｂを用いて暗号化し、暗号文Ｅ２を生成し、さらに、暗号文Ｅ２とＩＤ（ｂ）との排他的論理和を求め、ＤＥＳ暗号化部において、鍵Ｋａｂを用いて暗号化して生成した暗号文Ｅ３とによって送信データ（Token-AB）を生成する。

これを受信したＢは、受信データを、やはり共通の秘密鍵としてそれぞれの記録素子内に格納する鍵Ｋａｂ（認証キー）で復号化する。受信データの復号化方法は、まず、暗号文Ｅ１を認証キーＫａｂで復号化し、乱数Ｒａを得る。次に、暗号文Ｅ２を認証キーＫａｂで復号化し、その結果とＥ１の排他的論理和を求め、Ｒｂを得る。最後に、暗号文Ｅ３を認証キーＫａｂで復号化し、その結果とＥ２

の排他的論理和を求め、ＩＤ(ｂ)を得る。こうして得られたＲａ、Ｒｂ、ＩＤ(ｂ)の内、Ｒｂ及びＩＤ(ｂ)が、Ｂが送信したものと一致するか検証する。この検証に通った場合、ＢはＡを正当なものとして認証する。

次にＢは、認証後に使用するセッションキー（Ｋｓｅｓ）を生成する（生成方法は、乱数を用いる）。そして、Ｒｂ、Ｒａ、Ｋｓｅｓの順に、ＤＥＳのＣＢＣモードで認証キーＫａｂを用いて暗号化し、Ａに返送する。

これを受信したＡは、受信データを認証キーＫａｂで復号化する。受信データの復号化方法は、Ｂの復号化処理と同様であるので、ここでは詳細を省略する。こうして得られたＲｂ、Ｒａ、Ｋｓｅｓの内、Ｒｂ及びＲａが、Ａが送信したものと一致するか検証する。この検証に通った場合、ＡはＢを正当なものとして認証する。互いに相手を認証した後には、セッションキーＫｓｅｓは、認証後の秘密通信のための共通鍵として利用される。

なお、受信データの検証の際に、不正、不一致が見つかった場合には、相互認証が失敗したものとして処理を中断する。

上述の認証処理においては、Ａ，Ｂは共通の認証キーＫａｂを共有する。この共通鍵Ｋａｂを上述の有効化キーブロック（ＥＫＢ）を使用してデバイスに配信する。

例えば、図１２の例では、Ａ，又はＢのいずれかが他方が復号可能な有効化キーブロック（ＥＫＢ）を生成して生成した有効化キーブロック（ＥＫＢ）によって認証キーＫａｂを暗号化して、他方に送信する構成としてもよいし、あるいは第３者がデバイスＡ，Ｂに対して双方が利用可能な有効化キーブロック（ＥＫＢ）を生成してデバイスＡ，Ｂに対して生成した有効化キーブロック（ＥＫＢ）によって認証キーＫａｂを暗号化して配信する構成としてもよい。

図１３及び図１４に複数のデバイスに共通の認証キーＫａｋｅを有効化キーブロック（ＥＫＢ）によって配信する構成例を示す。図１３はデバイス０，１，２，３に対して復号可能な認証キーＫａｋｅを配信する例、図１４はデバイス０，１，２，３中のデバイス３をリボーク（排除）してデバイス０，１，２に対してのみ復号可能な認証キーを配信する例を示す。

図１３の例では、更新ノードキーＫ（ｔ）００によって、認証キーＫａｋｅを

暗号化したデータとともに、デバイス０，１，２，３においてそれぞれの有するノードキー、リーフキーを用いて更新されたノードキーＫ（ｔ）００を復号可能な有効化キーブロック（ＥＫＢ）を生成して配信する。それぞれのデバイスは、図１３の右側に示すようにまず、ＥＫＢを処理（復号）することにより、更新されたノードキーＫ（ｔ）００を取得し、次に、取得したノードキーＫ（ｔ）００を用いて暗号化された認証キー：Ｅｎｃ（Ｋ（ｔ）００，Ｋａｋｅ）を復号して認証キーＫａｋｅを得ることが可能となる。

その他のデバイス４，５，６，７…は同一の有効化キーブロック（ＥＫＢ）を受信しても自身の保有するノードキー、リーフキーでは、ＥＫＢを処理して更新されたノードキーＫ（ｔ）００を取得することができないので、安全に正当なデバイスに対してのみ認証キーを送付することができる。

一方、図１４の例は、図３の点線枠で囲んだグループにおいてデバイス３が、例えば鍵の漏洩によりリボーク（排除）されているとして、他のグループのメンバ、すなわち、デバイス０，１，２に対してのみ復号可能な有効化キーブロック（ＥＫＢ）を生成して配信した例である。図１４に示す有効化キーブロック（ＥＫＢ）と、認証キー（Ｋａｋｅ）をノードキー（Ｋ（ｔ）００）で暗号化したデータを配信する。

図１４の右側には、復号手順を示してある。デバイス０，１，２は、まず、受領した有効化キーブロックから自身の保有するリーフキー又はノードキーを用いた復号処理により、更新ノードキー（Ｋ（ｔ）００）を取得する。次に、Ｋ（ｔ）００による復号により認証キーＫａｋｅを取得する。

図３に示す他のグループのデバイス４，５，６…は、この同様のデータ（ＥＫＢ）を受信したとしても、自身の保有するリーフキー、ノードキーを用いて更新ノードキー（Ｋ（ｔ）００）を取得することができない。同様にリボークされたデバイス３においても、自身の保有するリーフキー、ノードキーでは、更新ノードキー（Ｋ（ｔ）００）を取得することができず、正当な権利を有するデバイスのみが認証キーを復号して利用することが可能となる。

このように、ＥＫＢを利用した認証キーの配送を用いれば、データ量を少なくして、かつ安全に正当権利者のみが復号可能とした認証キーを配信することが可

能となる。

［公開鍵認証と有効化キーブロック（ＥＫＢ）を使用したコンテンツキーの配信］

次に、公開鍵認証と有効化キーブロック（ＥＫＢ）を使用したコンテンツキーの配信処理について説明する。まず、公開鍵暗号方式である１６０ビット長の楕円曲線暗号を用いた相互認証方法を、図１５を用いて説明する。図１５において、公開鍵暗号方式としてＥＣＣを用いているが、同様な公開鍵暗号方式であればいずれでもよい。また、鍵サイズも１６０ビットでなくてもよい。図１５において、まずＢが、６４ビットの乱数Ｒｂを生成し、Ａに送信する。これを受信したＡは、新たに６４ビットの乱数Ｒａ及び素数ｐより小さい乱数Ａｋを生成する。そして、ベースポイントＧをＡｋ倍した点Ａｖ＝Ａｋ×Ｇを求め、Ｒａ、Ｒｂ、Ａｖ（Ｘ座標とＹ座標）に対する電子署名Ａ．Ｓｉｇを生成し、Ａの公開鍵証明書とともにＢに返送する。ここで、Ｒａ及びＲｂはそれぞれ６４ビット、ＡｖのＸ座標とＹ座標がそれぞれ１６０ビットであるので、合計４４８ビットに対する電子署名を生成する。

Ａの公開鍵証明書、Ｒａ、Ｒｂ、Ａｖ、電子署名Ａ．Ｓｉｇを受信したＢは、Ａが送信してきたＲｂが、Ｂが生成したものと一致するか検証する。その結果、一致していた場合には、Ａの公開鍵証明書内の電子署名を認証局の公開鍵で検証し、Ａの公開鍵を取り出す。そして、取り出したＡの公開鍵を用い電子署名Ａ．Ｓｉｇを検証する。

次に、Ｂは、素数ｐより小さい乱数Ｂｋを生成する。そして、ベースポイントＧをＢｋ倍した点Ｂｖ＝Ｂｋ×Ｇを求め、Ｒｂ、Ｒａ、Ｂｖ（Ｘ座標とＹ座標）に対する電子署名Ｂ．Ｓｉｇを生成し、Ｂの公開鍵証明書とともにＡに返送する。

Ｂの公開鍵証明書、Ｒｂ、Ｒａ、Ａｖ、電子署名Ｂ．Ｓｉｇを受信したＡは、Ｂが送信してきたＲａが、Ａが生成したものと一致するか検証する。その結果、一致していた場合には、Ｂの公開鍵証明書内の電子署名を認証局の公開鍵で検証し、Ｂの公開鍵を取り出す。そして、取り出したＢの公開鍵を用い電子署名Ｂ．Ｓｉｇを検証する。電子署名の検証に成功した後、ＡはＢを正当なものとして認証する。

両者が認証に成功した場合には、ＢはＢｋ×Ａｖ（Ｂｋは乱数だが、Ａｖは楕円曲線上の点であるため、楕円曲線上の点のスカラー倍計算が必要）を計算し、ＡはＡｋ×Ｂｖを計算し、これら点のＸ座標の下位６４ビットをセッションキーとして以降の通信に使用する（共通鍵暗号を６４ビット鍵長の共通鍵暗号とした場合）。もちろん、Ｙ座標からセッション鍵を生成してもよいし、下位６４ビットでなくてもよい。なお、相互認証後の秘密通信においては、送信データはセッションキーで暗号化されるだけでなく、電子署名も付されることがある。

電子署名の検証や受信データの検証の際に、不正、不一致が見つかった場合には、相互認証が失敗したものとして処理を中断する。

図１６に公開鍵認証と有効化キーブロック（ＥＫＢ）を使用したコンテンツキーの配信処理例を示す。まずコンテンツプロバイダとＰＣ間において図１５で説明した公開鍵方式による認証処理が実行される。コンテンツプロバイダは、コンテンツキー配信先である再生装置、記録媒体の有するノードキー、リーフキーによって復号可能なＥＫＢを生成して、更新ノードキーによる暗号化を実行したコンテンツキーＥ（Ｋｃｏｎ）と、有効化キーブロック（ＥＫＢ）とをＰＣ間の認証処理において生成したセッションキーＫｓｅｓで暗号化してＰＣに送信する。

ＰＣはセッションキーで暗号化された［更新ノードキーによる暗号化を実行したコンテンツキーＥ（Ｋｃｏｎ）と、有効化キーブロック（ＥＫＢ）］をセッションキーで復号した後、再生装置、記録媒体に送信する。

再生装置、記録媒体は、自身の保有するノードキー又はリーフキーによって［更新ノードキーによる暗号化を実行したコンテンツキーＥ（Ｋｃｏｎ）と、有効化キーブロック（ＥＫＢ）］を復号することによってコンテンツキーＫｃｏｎを取得する。

この構成によれば、コンテンツプロバイダとＰＣ間での認証を条件として［更新ノードキーによる暗号化を実行したコンテンツキーＥ（Ｋｃｏｎ）と、有効化キーブロック（ＥＫＢ）］が送信されるので、例えば、ノードキーの漏洩があった場合でも、確実な相手に対するデータ送信が可能となる。

［プログラムコードの有効化キーブロック（ＥＫＢ）を使用した配信］

上述した例では、コンテンツキー、認証キー等を有効化キーブロック（ＥＫ

Ｂ）を用いて暗号化して配信する方法を説明したが、様々なプログラムコードを有効化キーブロック（ＥＫＢ）を用いて配信する構成も可能である。すなわちＥＫＢによる暗号化メッセージデータをプログラムコードとした例である。以下、この構成について説明する。

図１７にプログラムコードを有効化キーブロック（ＥＫＢ）の例えば更新ノードキーによって暗号化してデバイス間で送信する例を示す。デバイス１７０１は、デバイス１７０２の有するノードキー、リーフキーによって復号可能な有効化キーブロック（ＥＫＢ）と、有効化キーブロック（ＥＫＢ）に含まれる更新ノードキーで暗号処理したプログラムコードをデバイス１７０２に送信する。デバイス１７０２は受信したＥＫＢを処理して更新ノードキーを取得して、さらに取得した更新ノードキーによってプログラムコードの復号を実行して、プログラムコードを得る。

図１７に示す例では、さらに、デバイス１７０２において取得したプログラムコードによる処理を実行して、その結果をデバイス１７０１に返して、デバイス１７０１がその結果に基づいて、さらに処理を続行する例を示している。

このように有効化キーブロック（ＥＫＢ）と、有効化キーブロック（ＥＫＢ）に含まれる更新ノードキーで暗号処理したプログラムコードを配信することにより、特定のデバイスにおいて解読可能なプログラムコードを前述の図３で示した特定のデバイス、あるいはグループに対して配信することが可能となる。

［送信コンテンツに対するチェック値（ＩＣＶ:Integrity Check Value）を対応させる構成］

次に、コンテンツの改竄を防止するためにコンテンツのインテグリティ・チェック値（ＩＣＶ）を生成して、コンテンツに対応付けて、ＩＣＶの計算により、コンテンツ改竄の有無を判定する処理構成について説明する。

コンテンツのインテグリティ・チェック値（ＩＣＶ）は、例えばコンテンツに対するハッシュ関数を用いて計算され、ＩＣＶ＝ｈａｓｈ（Ｋｉｃｖ，Ｃ１，Ｃ２，…）によって計算される。ＫｉｃｖはＩＣＶ生成キーである。Ｃ１，Ｃ２はコンテンツの情報であり、コンテンツの重要情報のメッセージ認証符号（ＭＡＣ：Message authentication Code）が使用される。

ＤＥＳ暗号処理構成を用いたＭＡＣ値生成例を図１８に示す。図１８の構成に示すように対象となるメッセージを８バイト単位に分割し、（以下、分割されたメッセージをＭ１、Ｍ２、・・・、ＭＮとする）、まず、初期値（Initial Value（以下、ＩＶとする））とＭ１の排他的論理和を求める（その結果をＩ１とする）。次に、Ｉ１をＤＥＳ暗号化部に入れ、鍵（以下、Ｋ１とする）を用いて暗号化する（出力をＥ１とする）。続けて、Ｅ１及びＭ２の排他的論理和を求め、その出力Ｉ２をＤＥＳ暗号化部へ入れ、鍵Ｋ１を用いて暗号化する（出力Ｅ２）。以下、これを繰り返し、全てのメッセージに対して暗号化処理を施す。最後に出てきたＥＮがメッセージ認証符号（ＭＡＣ（Message Authentication Code））となる。

このようなコンテンツのＭＡＣ値とＩＣＶ生成キーにハッシュ関数を適用して用いてコンテンツのインテグリティ・チェック値（ＩＣＶ）が生成される。改竄のないことが保証された例えばコンテンツ生成時に生成したＩＣＶと、新たにコンテンツに基づいて生成したＩＣＶとを比較して同一のＩＣＶが得られればコンテンツに改竄のないことが保証され、ＩＣＶが異なれば、改竄があったと判定される。

［チェック値（ＩＣＶ）の生成キーＫｉｃｖをＥＫＢによって配布する構成］

次に、コンテンツのインテグリティ・チェック値（ＩＣＶ）生成キーであるＫｉｃｖを上述の有効化キーブロックによって送付する構成について説明する。すなわちＥＫＢによる暗号化メッセージデータをコンテンツのインテグリティ・チェック値（ＩＣＶ）生成キーとした例である。

図１９及び図２０に複数のデバイスに共通のコンテンツを送付した場合、それらのコンテンツの改竄の有無を検証するためのインテグリティ・チェック値生成キーＫｉｃｖを有効化キーブロック（ＥＫＢ）によって配信する構成例を示す。図１９はデバイス０，１，２，３に対して復号可能なチェック値生成キーＫｉｃｖを配信する例、図２０はデバイス０，１，２，３中のデバイス３をリボーク（排除）してデバイス０，１，２に対してのみ復号可能なチェック値生成キーＫｉｃｖを配信する例を示す。

図１９の例では、更新ノードキーＫ（ｔ）００によって、チェック値生成キーＫｉｃｖを暗号化したデータとともに、デバイス０，１，２，３においてそれぞれの有するノードキー、リーフキーを用いて更新されたノードキーＫ（ｔ）００を復号可能な有効化キーブロック（ＥＫＢ）を生成して配信する。それぞれのデバイスは、図１９の右側に示すようにまず、ＥＫＢを処理（復号）することにより、更新されたノードキーＫ（ｔ）００を取得し、次に、取得したノードキーＫ（ｔ）００を用いて暗号化されたチェック値生成キー：Ｅｎｃ（Ｋ（ｔ）００，Ｋｉｃｖ）を復号してチェック値生成キーＫｉｃｖを得ることが可能となる。

その他のデバイス４，５，６，７…は同一の有効化キーブロック（ＥＫＢ）を受信しても自身の保有するノードキー、リーフキーでは、ＥＫＢを処理して更新されたノードキーＫ（ｔ）００を取得することができないので、安全に正当なデバイスに対してのみチェック値生成キーを送付することができる。

一方、図２０の例は、図３の点線枠で囲んだグループにおいてデバイス３が、例えば鍵の漏洩によりリボーク（排除）されているとして、他のグループのメンバ、すなわち、デバイス０，１，２に対してのみ復号可能な有効化キーブロック（ＥＫＢ）を生成して配信した例である。図２０に示す有効化キーブロック（ＥＫＢ）と、チェック値生成キー（Ｋｉｃｖ）をノードキー（Ｋ（ｔ）００）で暗号化したデータを配信する。

図２０の右側には、復号手順を示してある。デバイス０，１，２は、まず、受領した有効化キーブロックから自身の保有するリーフキー又はノードキーを用いた復号処理により、更新ノードキー（Ｋ（ｔ）００）を取得する。次に、Ｋ（ｔ）００による復号によりチェック値生成キーＫｉｃｖを取得する。

図３に示す他のグループのデバイス４，５，６…は、この同様のデータ（ＥＫＢ）を受信したとしても、自身の保有するリーフキー、ノードキーを用いて更新ノードキー（Ｋ（ｔ）００）を取得することができない。同様にリボークされたデバイス３においても、自身の保有するリーフキー、ノードキーでは、更新ノードキー（Ｋ（ｔ）００）を取得することができず、正当な権利を有するデバイスのみがチェック値生成キーを復号して利用することが可能となる。

このように、ＥＫＢを利用したチェック値生成キーの配送を用いれば、データ

量を少なくして、かつ安全に正当権利者のみが復号可能としたチェック値生成キーを配信することが可能となる。

このようなコンテンツのインテグリティ・チェック値（ＩＣＶ）を用いることにより、ＥＫＢと暗号化コンテンツの不正コピーを排除することができる。例えば図２１Ａ及び図２１Ｂに示すように、コンテンツＣ１とコンテンツＣ２とをそれぞれのコンテンツキーを取得可能な有効化キーブロック（ＥＫＢ）とともに格納したメディア１があり、これをそのままメディア２にコピーした場合を想定する。ＥＫＢと暗号化コンテンツのコピーは可能であり、これをＥＫＢを復号可能なデバイスでは利用できることになる。

図２１Ｂに示すように各メディアに正当に格納されたコンテンツに対応付けてインテグリティ・チェック値（ＩＣＶ（Ｃ１，Ｃ２））を格納する構成とする。なお、（ＩＣＶ（Ｃ１，Ｃ２））は、コンテンツＣ１とコンテンツＣ２にハッシュ関数を用いて計算されるコンテンツのインテグリティ・チェック値であるＩＣＶ＝ｈａｓｈ（Ｋｉｃｖ，Ｃ１，Ｃ２）を示している。図２１Ｂの構成において、メディア１には正当にコンテンツ１とコンテンツ２が格納され、コンテンツＣ１とコンテンツＣ２に基づいて生成されたインテグリティ・チェック値（ＩＣＶ（Ｃ１，Ｃ２））が格納される。また、メディア２には正当にコンテンツ１が格納され、コンテンツＣ１に基づいて生成されたインテグリティ・チェック値（ＩＣＶ（Ｃ１））が格納される。この構成において、メディア１に格納された｛ＥＫＢ，コンテンツ２｝をメディア２にコピーしたとすると、メディア２で、コンテンツチェック値を新たに生成するとＩＣＶ（Ｃ１，Ｃ２）が生成されることになり、メディアに格納されているＫｉｃｖ（Ｃ１）と異なり、コンテンツの改竄あるいは不正なコピーによる新たなコンテンツの格納が実行されたことが明らかになる。メディアを再生するデバイスにおいて、再生ステップの前ステップにＩＣＶチェックを実行して、生成ＩＣＶと格納ＩＣＶの一致を判別し、一致しない場合は、再生を実行しない構成とすることにより、不正コピーのコンテンツの再生を防止することが可能となる。

また、さらに、安全性を高めるため、コンテンツのインテグリティ・チェック値（ＩＣＶ）を書き換えカウンタを含めたデータに基づいて生成する構成として

もよい。すなわちＩＣＶ＝ｈａｓｈ（Ｋｉｃｖ，ｃｏｕｎｔｅｒ＋１，Ｃ１，Ｃ２，…）によって計算する構成とする。ここで、カウンタ（ｃｏｕｎｔｅｒ＋１）は、ＩＣＶの書き換え毎に１つインクリメントされる値として設定する。なお、カウンタ値はセキュアなメモリに格納する構成とすることが必要である。

さらに、コンテンツのインテグリティ・チェック値（ＩＣＶ）をコンテンツと同一メディアに格納することができない構成においては、コンテンツのインテグリティ・チェック値（ＩＣＶ）をコンテンツとは別のメディア上に格納する構成としてもよい。

例えば、読み込み専用メディアや通常のＭＯ等のコピー防止策のとられていないメディアにコンテンツを格納する場合、同一メディアにインテグリティ・チェック値（ＩＣＶ）を格納するとＩＣＶの書き換えが不正なユーザによりなされる可能性があり、ＩＣＶの安全性が保てないおそれがある。このような場合、ホストマシン上の安全なメディアにＩＣＶを格納して、コンテンツのコピーコントロール（例えばcheck-in/check-out、move）にＩＣＶを使用する構成とすることにより、ＩＣＶの安全な管理及びコンテンツの改竄チェックが可能となる。

この構成例を図２２に示す。図２２では読み込み専用メディアや通常のＭＯ等のコピー防止策のとられていないメディア２２０１にコンテンツが格納され、これらのコンテンツに関するインテグリティ・チェック値（ＩＣＶ）を、ユーザが自由にアクセスすることの許可されないホストマシン上の安全なメディア２２０２に格納し、ユーザによる不正なインテグリティ・チェック値（ＩＣＶ）の書き換えを防止した例である。このような構成として、例えばメディア２２０１を装着したデバイスがメディア２２０１の再生を実行する際にホストマシンであるＰＣ、サーバにおいてＩＣＶのチェックを実行して再生の可否を判定する構成とすれば、不正なコピーコンテンツあるいは改竄コンテンツの再生を防止できる。

［階層ツリー構造のカテゴリ分類］

暗号鍵をルートキー、ノードキー、リーフキー等、図３の階層ツリー構造として構成し、コンテンツキー、認証キー、ＩＣＶ生成キー、あるいはプログラムコード、データ等を有効化キーブロック（ＥＫＢ）とともに暗号化して配信する構成について説明してきたが、ノードキー等を定義している階層ツリー構造を各デ

バイスのカテゴリ毎に分類して効率的なキー更新処理、暗号化キー配信、データ配信を実行する構成について、以下説明する。

図２３に階層ツリー構造のカテゴリの分類の一例を示す。図２３において、階層ツリー構造の最上段には、ルートキーＫｒｏｏｔ２３０１が設定され、以下の中間段にはノードキー２３０２が設定され、最下段には、リーフキー２３０３が設定される。各デバイスは個々のリーフキーと、リーフキーからルートキーに至る一連のノードキー、ルートキーを保有する。

ここで、一例として最上段から第Ｍ段目のあるノードをカテゴリノード２３０４として設定する。すなわち第Ｍ段目のノードの各々を特定カテゴリのデバイス設定ノードとする。第Ｍ段の１つのノードを頂点として以下、Ｍ＋１段以下のノード、リーフは、そのカテゴリに含まれるデバイスに関するノード及びリーフとする。

例えば図２３の第Ｍ段目の１つのノード２３０５にはカテゴリ［メモリスティック（商標）］が設定され、このノード以下に連なるノード、リーフはメモリスティックを使用した様々なデバイスを含むカテゴリ専用のノード又はリーフとして設定される。すなわち、ノード２３０５以下を、メモリスティックのカテゴリに定義されるデバイスの関連ノード、及びリーフの集合として定義する。

さらに、Ｍ段から数段分下位の段をサブカテゴリノード２３０６として設定することができる。例えば図に示すようにカテゴリ［メモリスティック］ノード２３０５の２段下のノードに、メモリスティックを使用したデバイスのカテゴリに含まれるサブカテゴリノードとして、［再生専用器］のノードを設定する。さらに、サブカテゴリノードである再生専用器のノード２３０６以下に、再生専用器のカテゴリに含まれる音楽再生機能付き電話のノード２３０７が設定され、さらにその下位に、音楽再生機能付き電話のカテゴリに含まれる［ＰＨＳ］ノード２３０８と［携帯電話］ノード２３０９を設定することができる。

さらに、カテゴリ、サブカテゴリは、デバイスの種類のみならず、例えばあるメーカー、コンテンツプロバイダ、決済機関等が独自に管理するノード、すなわち処理単位、管轄単位、あるいは提供サービス単位等、任意の単位（これらを総称して以下、エンティティと呼ぶ）で設定することが可能である。例えば１つの

カテゴリノードをゲーム機器メーカーの販売するゲーム機器ＸＹＺ専用の頂点ノードとして設定すれば、メーカーの販売するゲーム機器ＸＹＺにその頂点ノード以下の下段のノードキー、リーフキーを格納して販売することが可能となり、その後、暗号化コンテンツの配信、あるいは各種キーの配信、更新処理を、その頂点ノードキー以下のノードキー、リーフキーによって構成される有効化キーブロック（ＥＫＢ）を生成して配信し、頂点ノード以下のデバイスに対してのみ利用可能なデータが配信可能となる。

このように、１つのノードを頂点としして、以下のノードをその頂点ノードに定義されたカテゴリ、あるいはサブカテゴリの関連ノードとして設定する構成とすることにより、カテゴリ段、あるいはサブカテゴリ段の１つの頂点ノードを管理するメーカー、コンテンツプロバイダ等がそのノードを頂点とする有効化キーブロック（ＥＫＢ）を独自に生成して、頂点ノード以下に属するデバイスに配信する構成が可能となり、頂点ノードに属さない他のカテゴリのノードに属するデバイスには全く影響を及ぼさずにキー更新を実行することができる。

［簡略ＥＫＢによるキー配信構成（１）］

先に説明した例えば図３のツリー構成において、キー、例えばコンテンツキーを所定デバイス（リーフ）宛に送付する場合、キー配布先デバイスの所有しているリーフキー、ノードキーを用いて復号可能な有効化キーブロック（ＥＫＢ）を生成して提供する。例えば図２４Ａに示すツリー構成において、リーフを構成するデバイスａ，ｇ，ｊに対してキー、例えばコンテンツキーを送信する場合、ａ，ｇ，ｊの各ノードにおいて復号可能な有効化キーブロック（ＥＫＢ）を生成して配信する。

例えば更新ルートキーＫ（ｔ）ｒｏｏｔでコンテンツキーＫ（ｔ）ｃｏｎを暗号化処理し、ＥＫＢとともに配信する場合を考える。この場合、デバイスａ，ｇ，ｊは、それぞれが図２４Ｂに示すリーフ及びノードキーを用いて、ＥＫＢの処理を実行してＫ（ｔ）ｒｏｏｔを取得し、取得した更新ルートキーＫ（ｔ）ｒｏｏｔによってコンテンツキーＫ（ｔ）ｃｏｎの復号処理を実行してコンテンツキーを得る。

この場合に提供される有効化キーブロック（ＥＫＢ）の構成は、図２５に示す

ようになる。図２５に示す有効化キーブロック（ＥＫＢ）は、先の図６で説明した有効化キーブロック（ＥＫＢ）のフォーマットにしたがって構成されたものであり、データ（暗号化キー）と対応するタグとを持つ。タグは、先に図７Ａ乃至図７Ｃを用いて説明したように左（Ｌ）、右（Ｒ）、それぞれの方向にデータがあれば０、無ければ１を示している。

有効化キーブロック（ＥＫＢ）を受領したデバイスは、有効化キーブロック（ＥＫＢ）の暗号化キーとタグに基づいて、順次暗号化キーの復号処理を実行して上位ノードの更新キーを取得していく。図２５に示すように、有効化キーブロック（ＥＫＢ）は、ルートからリーフまでの段数（デプス）が多いほど、そのデータ量は増加していく。段数（デプス）は、デバイス（リーフ）数に応じて増大するものであり、キーの配信先となるデバイス数が多い場合は、ＥＫＢのデータ量がさらに増大することになる。

このような有効化キーブロック（ＥＫＢ）のデータ量の削減を可能とした構成について説明する。図２６Ａ及び図２６Ｂは、有効化キーブロック（ＥＫＢ）をキー配信デバイスに応じて簡略化して構成した例を示すものである。

図２５と同様、リーフを構成するデバイスａ，ｇ，ｊに対してキー、例えばコンテンツキーを送信する場合を想定する。図２６Ａに示すように、キー配信デバイスによってのみ構成されるツリーを構築する。この場合、図２４Ｂに示す構成に基づいて新たなツリー構成として図２６Ｂのツリー構成が構築される。ＫｒｏｏｔからＫｊまでは全く分岐がなく１つの枝のみが存在すればよく、ＫｒｏｏｔからＫａ及びＫｇに至るためには、Ｋ０に分岐点を構成するのみで、２分岐構成の図２６Ａのツリーが構築される。

図２６Ａに示すように、ノードとしてＫ０のみを持つ簡略化したツリーが生成される。更新キー配信のための有効化キーブロック（ＥＫＢ）は、これらの簡略ツリーに基づいて生成する。図２６Ａに示すツリーは、有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段とした２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーである。更新キー配信のための有効化キーブロック（ＥＫＢ）は、この再構築階層ツリーのノード又はリーフに対応するキーのみに基づいて構成される。

先の図２５で説明した有効化キーブロック（ＥＫＢ）は、各リーフａ，ｇ，ｊからＫｒｏｏｔに至るまでの全てのキーを暗号化したデータを格納していたが、簡略化ＥＫＢは、簡略化したツリーを構成するノードについてのみの暗号化データを格納する。図２６Ｂに示すようにタグは３ビット構成を有する。第１及び第２ビットは、図２５の例と、同様の意味を持ち、左（Ｌ）、右（Ｒ）、それぞれの方向にデータがあれば０、無ければ１を示す。第３番目のビットは、ＥＫＢ内に暗号化キーが格納されているか否かを示すためのビットであり、データが格納されている場合は１、データが無い場合は、０として設定される。

データ通信網、あるいは記憶媒体に格納されてデバイス（リーフ）に提供される有効化キーブロック（ＥＫＢ）は、図２６Ｂに示すように、図２５に示す構成に比較すると、データ量が大幅に削減されたものとなる。図２６Ａ及び図２６Ｂに示す有効化キーブロック（ＥＫＢ）を受領した各デバイスは、タグの第３ビットに１が格納された部分のデータのみを順次復号することにより、所定の暗号化キーの復号を実現することができる。例えばデバイスａは、暗号化データＥｎｃ（Ｋａ，Ｋ（ｔ）０）をリーフキーＫａで復号して、ノードキーＫ（ｔ）０を取得して、ノードキーＫ（ｔ）０によって暗号化データＥｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）ｒｏｏｔ）を復号してＫ（ｔ）ｒｏｏｔを取得する。デバイスｊは、暗号化データＥｎｃ（Ｋｊ，Ｋ（ｔ）ｒｏｏｔ）をリーフキーＫｊで復号して、Ｋ（ｔ）ｒｏｏｔを取得する。

このように、配信先のデバイスによってのみ構成される簡略化した新たなツリー構成を構築して、構築されたツリーを構成するリーフ及びノードのキーのみを用いて有効化キーブロック（ＥＫＢ）を生成することにより、少ないデータ量の有効化キーブロック（ＥＫＢ）を生成することが可能となり、有効化キーブロック（ＥＫＢ）のデータ配信が効率的に実行可能となる。

［簡略ＥＫＢによるキー配信構成（２）］

図２６Ａ及び図２６Ｂで示した簡略化したツリーに基づいて生成される有効化キーブロック（ＥＫＢ）をさらに、簡略化してデータ量を削減し、効率的な処理を可能とした構成について説明する。

図２６Ａ及び図２６Ｂを用いて説明した構成は、有効化キーブロック（ＥＫ

Ｂ）を復号可能な末端ノード又はリーフを最下段とした２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーであった。更新キー配信のための有効化キーブロック（ＥＫＢ）は、この再構築階層ツリーのノード又はリーフに対応するキーのみに基づいて構成される。

図２６Ａに示す再構築階層ツリーは、リーフａ，ｇ，ｊにおいて更新ルートキーＫ（ｔ）ｒｏｏｔを取得可能とするため、図２６Ｂに示す有効化キーブロック（ＥＫＢ）を配信する。図２６Ｂの有効化キーブロック（ＥＫＢ）の処理において、リーフｊは、Ｅｎｃ（Ｋｊ，Ｋ（ｔ）ｒｏｏｔ）の１回の復号処理によりルートキー：Ｋ（ｔ）ｒｏｏｔを取得できる。しかし、リーフａ，ｇは、Ｅｎｃ（Ｋａ，Ｋ（ｔ）０）又は、Ｅｎｃ（Ｋｇ，Ｋ（ｔ）０）の復号処理によりＫ（ｔ）０を得た後、さらに、Ｅｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）ｒｏｏｔ）の復号処理を実行してルートキー：Ｋ（ｔ）ｒｏｏｔを取得する。すなわち、リーフａ，ｇは、２回の復号処理を実行することが必要となる。

図２６Ａ及び図２６Ｂの簡略化した再構築階層ツリーは、ノードＫ０がその下位リーフａ，ｇの管理ノードとして独自の管理を実行している場合、例えば後述するサブルート・ノードとして、下位リーフの管理を実行している場合には、リーフａ，ｇが更新キーを取得したことを確認する意味で有効であるが、ノードＫ０が下位リーフの管理を行なっていない場合、あるいは行なっていたとしても、上位ノードからの更新キー配信を許容している場合には、図２６Ａに示す再構築階層ツリーをさらに簡略化して、ノードＫ０のキーを省略して有効化キーブロック（ＥＫＢ）を生成して配信してもよい。

図２７Ａ及び図２７Ｂに、このような有効化キーブロック（ＥＫＢ）の構成を示す。図２６Ａ及び図２６Ｂと同様、リーフを構成するデバイスａ，ｇ，ｊに対してキー、例えばコンテンツキーを送信する場合を想定する。図２７Ａに示すように、ルートＫｒｏｏｔと各リーフａ，ｇ，ｊを直接接続したツリーを構築する。

図２７Ａに示すように、図２６Ａに示す再構築階層ツリーからノードＫ０が省かれた簡略化したツリーが生成される。更新キー配信のための有効化キーブロック（ＥＫＢ）は、これらの簡略ツリーに基づいて生成する。図２７Ａに示すツリーは、有効化キーブロック（ＥＫＢ）を復号可能なリーフをとルートとを直接結

ぶパスのみによって再構築される再構築階層ツリーである。更新キー配信のための有効化キーブロック（ＥＫＢ）は、この再構築階層ツリーのリーフに対応するキーのみに基づいて構成される。

なお、図２７Ａの例は、末端をリーフとした構成例であるが、例えば最上位ノードか複数の中位、下位ノードに対してキーを配信する場合も、最上位ノードと中位、下位ノードとを直接接続した簡略化ツリーに基づいて有効化キーブロック（ＥＫＢ）を生成してキー配信を実行することが可能である。このように、再構築階層ツリーは、簡略化したツリーを構成する頂点ノードと、簡略化したツリーを構成する末端ノード又はリーフとを直接、接続した構成を持つ。この簡略化ツリーでは、頂点ノードからの分岐は２に限らず、配信ノード又はリーフ数に応じて３以上の多分岐を持つツリーとして構成することが可能である。

先の図２５で説明した有効化キーブロック（ＥＫＢ）は、各リーフａ，ｇ，ｊからＫｒｏｏｔに至るまでの全てのキーを暗号化したデータを格納し、図２６Ａ及び図２６Ｂで説明した有効化キーブロック（ＥＫＢ）は、リーフａ，ｇ，ｊのリーフキー、ａ，ｇの共通ノードとしてのＫ０、さらに、ルートキーを格納した構成であったが、図２７Ａに示す簡略化階層ツリーに基づく有効化キーブロック（ＥＫＢ）は、ノードＫ０のキーを省略したので、図２７Ｂに示すように、さらにデータ量の少ない有効化キーブロック（ＥＫＢ）となる。

図２７Ｂの有効化キーブロック（ＥＫＢ）は、図２６Ｂの有効化キーブロック（ＥＫＢ）と同様、３ビット構成のタグを有する。第１及び第２ビットは、図２６Ａ及び図２６Ｂで説明したと同様、左（Ｌ）、右（Ｒ）、それぞれの方向にデータがあれば０、無ければ１を示す。第３番目のビットは、ＥＫＢ内に暗号化キーが格納されているか否かを示すためのビットであり、データが格納されている場合は１、データが無い場合は、０として設定される。

図２７Ｂの有効化キーブロック（ＥＫＢ）において、各リーフａ，ｇ，ｊは、Ｅｎｃ（Ｋａ，Ｋ（ｔ）ｒｏｏｔ）、又はＥｎｃ（Ｋｇ，Ｋ（ｔ）ｒｏｏｔ）Ｅｎｃ（Ｋｊ，Ｋ（ｔ）ｒｏｏｔ）の１回の復号処理によりルートキー：Ｋ（ｔ）ｒｏｏｔを取得できる。

このように簡略化された再構築ツリーの最上位ノードと、ツリーを構成する末

端ノード又はリーフとを直接、接続した構成を持つツリーに基づいて生成される有効化キーブロック（ＥＫＢ）は、図２７Ｂに示すように、再構築階層ツリーの頂点ノード及び末端ノード又はリーフに対応するキーのみに基づいて構成される。

図２６Ａ及び図２６Ｂ又は図２７Ａ及び図２７Ｂで説明した有効化キーブロック（ＥＫＢ）のように、配信先のデバイスによってのみ構成される簡略化した新たなツリー構成を構築して、構築されたツリーを構成するリーフのみ、あるいはリーフと共通ノードのキーのみを用いて有効化キーブロック（ＥＫＢ）を生成することにより、少ないデータ量の有効化キーブロック（ＥＫＢ）を生成することが可能となり、有効化キーブロック（ＥＫＢ）のデータ配信が効率的に実行可能となる。

なお、簡略化した階層ツリー構成は、後段で説明するエンティティ単位のＥＫＢ管理構成において特に有効に活用可能である。エンティティは、キー配信構成としてのツリー構成を構成するノードあるいはリーフから選択した複数のノードあるいはリーフの集合体ブロックである。エンティティは、デバイスの種類に応じて設定される集合であったり、あるいはデバイス提供メーカー、コンテンツプロバイダ、決済機関等の管理単位等、ある共通点を持った処理単位、管轄単位、あるいは提供サービス単位等、様々な態様の集合として設定される。１つのエンティティには、ある共通のカテゴリに分類されるデバイスが集まっており、例えば複数のエンティティの頂点ノード（サブルート）によって上述したと同様の簡略化したツリーを再構築してＥＫＢを生成することにより、選択されたエンティティに属するデバイスにおいて復号可能な簡略化された有効化キーブロック（ＥＫＢ）の生成、配信が可能となる。エンティティ単位の管理構成については後段で詳細に説明する。

なお、このような有効化キーブロック（ＥＫＢ）は、光ディスク、ＤＶＤ等の情報記録媒体に格納した構成とすることが可能である。例えば、上述の暗号化キーデータによって構成されるデータ部と、暗号化キーデータの階層ツリー構造における位置識別データとしてのタグ部とを含む有効化キーブロック（ＥＫＢ）にさらに、更新ノードキーによって暗号化したコンテンツ等のメッセージデータとを格納した情報記録媒体を各デバイスに提供する構成が可能である。デバイスは

有効化キーブロック（ＥＫＢ）に含まれる暗号化キーデータをタグ部の識別データにしたがって順次抽出して復号し、コンテンツの復号に必要なキーを取得してコンテンツの利用を行うことが可能となる。もちろん、有効化キーブロック（ＥＫＢ）をインターネット等のネットワークを介して配信する構成としてもよい。

［エンティティ単位のＥＫＢ管理構成］

次に、キー配信構成としてのツリー構成を構成するノードあるいはリーフを、複数のノードあるいはリーフの集合としてのブロックで管理する構成について説明する。なお、複数のノードあるいはリーフの集合としてのブロックを以下エンティティと呼ぶ。エンティティは、デバイスの種類に応じて設定される集合であったり、あるいはデバイス提供メーカー、コンテンツプロバイダ、決済機関等の管理単位等、ある共通点を持った処理単位、管轄単位、あるいは提供サービス単位等、様々な態様の集合として設定される。

エンティティについて、図２８乃至図２８Ｃを用いて説明する。図２８Ａはツリーのエンティティ単位での管理構成を説明する図である。１つのエンティティは図では、三角形として示し、例えば１エンティティ２７０１内には、複数のノードが含まれる。１エンティティ内のノード構成を示すのが図２８Ｂである。１つのエンティティは、１つのノードを頂点とした複数段の２分岐形ツリーによって構成される。以下、エンティティの頂点ノード２７０２をサブルートと呼ぶ。

ツリーの末端は、図２８Ｃに示すようにリーフ、すなわちデバイスによって構成される。デバイスは、複数デバイスをリーフとし、サブルートである頂点ノード２７０２を持つツリーによって構成されるいずれかのエンティティに属する。

図２８Ａから理解されるように、エンティティは、階層構造を持つ。この階層構造について、図２９Ａ乃至図２９Ｃを用いて説明する。

図２９Ａは、階層構造を簡略化して説明するための図であり、Ｋｒｏｏｔから数段下の段にエンティティＡ０１～Ａｎｎが構成され、エンティティＡ１～Ａｎの下位には、さらに、エンティティＢ０１～Ｂｎｋ、さらに、その下位にエンティティＣ１～Ｃｎｑが設定されている。各エンティティは、図２９Ｂ，図２９Ｃに示す如く、複数段のノード、リーフによって構成されるツリー形状を持つ。

例えばエンティティＢｎｋの構成は、（ｂ）に示すように、サブルート２８１

１を頂点ノードとして、末端ノード２８１２に至るまでの複数ノードを有する。このエンティティは識別子Ｂｎｋを持ち、エンティティＢｎｋ内のノードに対応するノードキー管理をエンティティＢｎｋ独自に実行することにより、末端ノード２８１２を頂点として設定される下位（子）エンティティの管理を実行する。また、一方、エンティティＢｎｋは、サブルート２８１１を末端ノードとして持つ上位（親）エンティティＡｎｎの管理下にある。

エンティティＣｎ３の構成は、（ｃ）に示すように、サブルート２８５１を頂点ノードとして、各デバイスである末端ノード２８５２、この場合はリーフに至るまで複数ノード、リーフを有する。このエンティティは識別子Ｃｎ３を持ち、エンティティＣｎ３内のノード、リーフに対応するノードキー、リーフキー管理をエンティティＣｎ３独自に実行することにより、末端ノード２８５２に対応するリーフ（デバイス）の管理を実行する。また、一方、エンティティＣｎ３は、サブルート２８５１を末端ノードとして持つ上位（親）エンティティＢｎ２の管理下にある。各エンティティにおけるキー管理とは、例えばキー更新処理、リボーク処理等であるが、これらは後段で詳細に説明する。

最下段エンティティのリーフであるデバイスには、デバイスの属するエンティティのリーフキーから、自己の属するエンティティの頂点ノードであるサブルートノードに至るパスに位置する各ノードのノードキー及びリーフキーが格納される。例えば末端ノード２８５２のデバイスは、末端ノード（リーフ）２８５２から、サブルートノード２８５１までの各キーを格納する。

図３０Ａ及び図３０Ｂを用いて、さらにエンティティの構成について説明する。エンティティは様々な段数によって構成されるツリー構造を持つことが可能である。段数、すなわちデプス（depth）は、エンティティで管理する末端ノードに対応する下位（子）エンティティの数、あるいはリーフとしてのデバイス数に応じて設定可能である。

図３０Ａに示すような上下エンティティ構成を具体化すると、（ｂ）に示す態様となる。ルートエンティティは、ルートキーを持つ最上段のエンティティである。ルートエンティティの末端ノードに複数の下位エンティティとしてエンティティＡ，Ｂ，Ｃが設定され、さらに、エンティティＣの下位エンティティとして

エンティティＤが設定される。エンティティＣ２９０１は、その末端ノードの１つ以上のノードをリザーブノード２９５０として保持し、自己の管理するエンティティを増加させる場合、さらに複数段のツリー構成を持つエンティティＣ’２９０２をリザーブノード２９５０を頂点ノードとして新設することにより、管理末端ノード２９７０を増加させて、管理末端ノードに増加した下位エンティティを追加することができる。

リザーブノードについて、さらに図３１を用いて説明する。エンティティＡ，３０１１は、管理する下位エンティティＢ，Ｃ，Ｄ…を持ち、１つのリザーブノード３０２１を持つ。エンティティは管理対象の下位エンティティをさらに増加させたい場合、リザーブノード３０２１に、自己管理の下位エンティティＡ’，３０１２を設定し、下位エンティティＡ’，３０１２の末端ノードにさらに管理対象の下位エンティティＦ，Ｇを設定することができる。自己管理の下位エンティティＡ’，３０１２も、その末端ノードの少なくとも１つをリザーブノード３０２２として設定することにより、さらに下位エンティティＡ’’３０１３を設定して、さらに管理エンティティを増加させることができる。下位エンティティＡ’’３０１３の末端ノードにも１以上のリザーブノードを確保する。このようなリザーブノード保有構成をとることにより、あるエンティティの管理する下位エンティティは、際限なく増加させることが可能となる。なお、リザーブエンティティは、末端ノードの１つのみではなく、複数個設定する構成としてもよい。

それぞれのエンティティでは、エンティティ単位で有効化キーブロック（ＥＫＢ）が構成され、エンティティ単位でのキー更新、リボーク処理を実行することになる。図３１のように複数のエンティティＡ，Ａ’，Ａ’’には各エンティティ個々の有効化キーブロック（ＥＫＢ）が設定されることになるが、これらは、エンティティＡ，Ａ’，Ａ’’を共通に管理する例えばあるデバイスメーカーが一括して管理することが可能である。

［新規エンティティの登録処理］

次に、新規エンティティの登録処理について説明する。登録処理シーケンスを図３２に示す。図３２のシーケンスにしたがって説明する。新たにツリー構成中に追加される新規（子）エンティティ（Ｎ－Ｅｎ）は、上位（親）エンティティ

（Ｐ－Ｅｎ）に対して新規登録要求を実行する。なお、各エンティティは、公開鍵暗号方式に従った公開鍵を保有し、新規エンティティは自己の公開鍵を登録要求に際して上位エンティティ（Ｐ－Ｅｎ）に送付する。

登録要求を受領した上位エンティティ（Ｐ－Ｅｎ）は、受領した新規（子）エンティティ（Ｎ－Ｅｎ）の公開鍵を証明書発行局（ＣＡ：Certificate Authority）に転送し、ＣＡの署名を付加した新規（子）エンティティ（Ｎ－Ｅｎ）の公開鍵を受領する。これらの手続きは、上位エンティティ（Ｐ－Ｅｎ）と新規（子）エンティティ（Ｎ－Ｅｎ）との相互認証の手続きとして行われる。

これらの処理により、新規登録要求エンティティの認証が終了すると、上位エンティティ（Ｐ－Ｅｎ）は、新規（子）エンティティ（Ｎ－Ｅｎ）の登録を許可し、新規（子）エンティティ（Ｎ－Ｅｎ）のノードキーを新規（子）エンティティ（Ｎ－Ｅｎ）に送信する。このノードキーは、上位エンティティ（Ｐ－Ｅｎ）の末端ノードの１つのノードキーであり、かつ、新規（子）エンティティ（Ｎ－Ｅｎ）の頂点ノード、すなわちサブルートキーに対応する。

このノードキー送信が終了すると、新規（子）エンティティ（Ｎ－Ｅｎ）は、新規（子）エンティティ（Ｎ－Ｅｎ）のツリー構成を構築し、構築したツリーの頂点に受信した頂点ノードのサブルートキーを設定し、各ノード、リーフのキーを設定して、エンティティ内の有効化キーブロック（ＥＫＢ）を生成する。１つのエンティティ内の有効化キーブロック（ＥＫＢ）をサブＥＫＢと呼ぶ。

一方、上位エンティティ（Ｐ－Ｅｎ）は、新規（子）エンティティ（Ｎ－Ｅｎ）の追加により、有効化する末端ノードを追加した上位エンティティ（Ｐ－Ｅｎ）内のサブＥＫＢを生成する。

新規（子）エンティティ（Ｎ－Ｅｎ）は、新規（子）エンティティ（Ｎ－Ｅｎ）内のノードキー、リーフキーによって構成されるサブＥＫＢを生成すると、これを上位エンティティ（Ｐ－Ｅｎ）に送信する。

新規（子）エンティティ（Ｎ－Ｅｎ）からサブＥＫＢを受信した上位エンティティ（Ｐ－Ｅｎ）は、受信したサブＥＫＢと、上位エンティティ（Ｐ－Ｅｎ）の更新したサブＥＫＢとをキー発行センタ（ＫＤＣ：Key Distribute Center）に送信する。

キー発行センタ（ＫＤＣ）は、全てのエンティティのサブＥＫＢに基づいて、様々な態様のＥＫＢ、すなわち特定のエンティティあるいはデバイスのみが復号可能なＥＫＢを生成することが可能となる。このように復号可能なエンティティあるいはデバイスを設定したＥＫＢを例えばコンテンツプロバイダに提供し、コンテンツプロバイダがＥＫＢに基づいてコンテンツキーを暗号化して、ネットワークを介して、あるいは記録媒体に格納して提供することにより、特定のデバイスでのみ利用可能なコンテンツを提供することが可能となる。

なお、新規エンティティのサブＥＫＢのキー発行センタ（ＫＤＣ）に対する登録処理は、サブＥＫＢを上位エンティティを介してを順次転送して実行する方法に限るものではなく、上位エンティティを介さずに、新規登録エンティティから直接、キー発行センタ（ＫＤＣ）に登録する処理を実行する構成としてもよい。

上位エンティティと、上位エンティティに新規追加する下位エンティティとの対応について図３３を用いて説明する。上位エンティティの末端ノードの１つ３２０１を新規追加エンティティの頂点ノードとして、下位エンティティに提供することによって下位エンティティは、上位エンティティの管理下のエンティティとして追加される。上位エンティティの管理下のエンティティとは、後段で詳細に説明するが、下位エンティティのリボーク（排除）処理を上位エンティティが実行できる構成であるという意味を含むものである。

図３３に示すように、上位エンティティに新規エンティティが設定されると、上位エンティティのリーフである末端ノードの１つのノード３２０１と新規追加エンティティの頂点ノード３２０２とが等しいノードとして設定される。すなわち上位ノードの１つのリーフである１つの末端ノードが、新規追加エンティティのサブルートとして設定される。このように設定されることにより、新規追加エンティティが全体ツリー構成の下で有効化される。

図３４Ａ及び図３４Ｂに新規追加エンティティを設定した際に上位エンティティが生成する更新ＥＫＢの例を示す。図３４Ａ及び図３４Ｂは、図３４Ａに示す構成、すなわち既に有効に存在する末端ノード（ｎｏｄｅ０００）３３０１と末端ノード（ｎｏｄｅ００１）３３０２があり、ここに新規追加エンティティに新規エンティティ追加末端ノード（ｎｏｄｅ１００）３３０３を付与した際に上位

エンティティが生成するサブＥＫＢの例を示したものである。

サブＥＫＢは、図３４Ｂに示すようにな構成を持つ。それぞれ有効に存在する末端ノードキーにより暗号化された上位ノードキー、上位ノードキーで暗号化されたさらなる上位ノードキー、…さらに上位に進行してサブルートキーに至る構成となっている。この構成によりサブＥＫＢが生成される。各エンティティは図３４Ｂに示すと同様、有効な末端ノード、あるいはリーフキーにより暗号化された上位ノードキー、上位ノードキーでさらに上位のノードキーを暗号化し、順次上位に深厚してサブルートに至る暗号化データによって構成されるＥＫＢを有し、これを管理する。

［エンティティ管理下におけるリボーク処理］

次に、キー配信ツリー構成をエンティティ単位として管理する構成におけるデバイスあるいはエンティティのリボーク（排除）処理について説明する。先の図３，４では、ツリー構成全体の中から特定のデバイスのみ復号可能で、リボークされたデバイスは復号不可能な有効化キーブロック（ＥＫＢ）を配信する処理について説明した。図３，４で説明したリボーク処理は、ツリー全体の中から特定のリーフであるデバイスをリボークする処理であったが、ツリーのエンティティ管理による構成では、エンティティ毎にリボーク処理が実行可能となる。

図３５以下の図を用いてエンティティ管理下のツリー構成におけるリボーク処理について説明する。図３５Ａ乃至図３５Ｄは、ツリーを構成するエンティティの内、最下段のエンティティ、すなわち個々のデバイスを管理しているエンティティによるデバイスのリボーク処理を説明する図である。

図３５Ａは、エンティティ管理によるキー配信ツリー構造を示している。ツリー最上位にはルートノードが設定され、その数段下にエンティティＡ０１～Ａｎｎ、さらにその下位段にＢ０１～Ｂｎｋのエンティティ、さらにその下位段にＣ１～ｃｎのエンティティが構成されている。最も下のエンティティは、末端ノード（リーフ）が個々のデバイス、例えば記録再生器、再生専用器等であるとする。

ここで、リボーク処理は、各エンティティにおいて独自に実行される。例えば、最下段のエンティティＣ１～Ｃｎでは、リーフのデバイスのリボーク処理が実行される。図３５Ｂには、最下段のエンティティの１つであるエンティティＣｎ，

３４３０のツリー構成を示している。エンティティＣｎ，３４３０は、頂点ノード３４３１を持ち、末端ノードであるリーフに複数のデバイスを持つ構成である。

この末端ノードであるリーフ中に、リボーク対象となるデバイス、例えばデバイス３４３２があったとすると、エンティティＣｎ，３４３０は、独自に更新したエンティティＣｎ内のノードキー、リーフキーによって構成される有効化キーブロック（サブＥＫＢ）を生成する。この有効化キーブロックは、リボークデバイス３４３２においては復号できず、他のリーフを構成するデバイスにおいてのみ復号可能な暗号化キーにより構成されるキーブロックである。エンティティＣｎの管理者は、これを更新されたサブＥＫＢとして生成する。具体的には、サブルートからリボークデバイス３４３２に連なるパスを構成する各ノード３４３１，３４３４，３４３５のノードキーを更新して、この更新ノードキーをリボークデバイス３４３２以外のリーフデバイスにおいてのみ復号可能な暗号化キーとして構成したブロックを更新サブＥＫＢとする。この処理は、先の図３，４において説明したリボーク処理構成において、ルートキーを、エンティティの頂点キーであるサブルートキーに置き換えた処理に対応する。

このようにエンティティＣｎ，３４３０がリボーク処理によって更新した有効化キーブロック（サブＥＫＢ）は、上位エンティティに送信される。この場合、上位エンティティはエンティティＢｎｋ，３４２０であり、エンティティＣｎ，３４３０の頂点ノード３４３１を末端ノードとして有するエンティティである。

エンティティＢｎｋ，３４２０は、下位エンティティＣｎ，３４３０から有効化キーブロック（サブＥＫＢ）を受領すると、そのキーブロックに含まれるエンティティＣｎｋ，３４３０の頂点ノード３４３１に対応するエンティティＢｎｋ，３４２０の末端ノード３４３１を、下位エンティティＣｎ，３４３０において更新されたキーに設定して、自身のエンティティＢｎｋ，３４２０のサブＥＫＢの更新処理を実行する。図３５Ｃにエンティティ Ｂｎｋ，３４２０のツリー構成を示す。エンティティＢｎｋ，３４２０において、更新対象となるノードキーは、図３５Ｃのサブルート３４２１からリボークデバイスを含むエンティティを構成する末端ノード３４３１に至るパス上のノードキーである。すなわち、更新サブＥＫＢを送信してきたエンティティのノード３４３１に連なるパスを構成する各

ノード３４２１，３４２４，３４２５のノードキーが更新対象となる。これら各ノードのノードキーを更新してエンティティＢｎｋ，３４２０の新たな更新サブＥＫＢを生成する。

さらに、エンティティＢｎｋ，３４２０が更新した有効化キーブロック（サブＥＫＢ）は、上位エンティティに送信される。この場合、上位エンティティはエンティティＡｎｎ，３４１０であり、エンティティＢｎｋ，３４２０の頂点ノード３４２１を末端ノードとして有するエンティティである。

エンティティＡｎｎ，３４１０は、下位エンティティＢｎｋ，３４２０から有効化キーブロック（サブＥＫＢ）を受領すると、そのキーブロックに含まれるエンティティＢｎｋ，３４２０の頂点ノード３４２１に対応するエンティティＡｎｎ，３４１０の末端ノード３４２１を、下位エンティティＢｎｋ，３４２０において更新されたキーに設定して、自身のエンティティＡｎｎ，３４１０のサブＥＫＢの更新処理を実行する。図３５Ｄにエンティティ Ａｎｎ，３４１０のツリー構成を示す。エンティティＡｎｎ，３４１０において、更新対象となるノードキーは、図３５Ｄのサブルート３４１１から更新サブＥＫＢを送信してきたエンティティのノード３４２１に連なるパスを構成する各ノード３４１１，３４１４，３４１５のノードキーである。これら各ノードのノードキーを更新してエンティティＡｎｎ，３４１０の新たな更新サブＥＫＢを生成する。

これらの処理を順次、上位のエンティティにおいて実行し、図３０Ｂで説明したルートエンティティまで実行する。この一連の処理により、デバイスのリボーク処理が完結する。なお、それぞれのエンティティにおいて更新されたサブＥＫＢは、最終的にキー発行センタ（ＫＤＣ）に送信され、保管される。キー発行センタ（ＫＤＣ）は、全てのエンティティの更新サブＥＫＢに基づいて、様々なＥＫＢを生成する。更新ＥＫＢは、リボークされたデバイスでの復号が不可能な暗号化キーブロックとなる。

デバイスのリボーク処理のシーケンス図を図３６に示す。処理手順を図３６のシーケンス図に従って説明する。まず、ツリー構成の最下段にあるデバイス管理エンティティ（Ｄ－Ｅｎ）は、デバイス管理エンティティ（Ｄ－Ｅｎ）内のリボーク対象のリーフを排除するために必要なキー更新を行ない、デバイス管理エン

ティティ（Ｄ－Ｅｎ）の新たなサブＥＫＢ（Ｄ）を生成する。更新サブＥＫＢ（Ｄ）は、上位エンティティに送付される。更新サブＥＫＢ（Ｄ）を受領した上位（親）エンティティ（Ｐ１－Ｅｎ）は、更新サブＥＫＢ（Ｄ）の更新頂点ノードに対応した末端ノードキーの更新及び、その末端ノードからサブルートに至るパス上のノードキーを更新した更新サブＥＫＢ（Ｐ１）を生成する。これらの処理を順次、上位エンティティにおいて実行して、最終的に更新された全てのサブＥＫＢがキー発行センタ（ＫＤＣ）に格納され管理される。

図３７Ａ及び図３７Ｂにデバイスのリボーク処理によって上位エンティティが更新処理を行なって生成する有効化キーブロック（ＥＫＢ）の例を示す。

図３７Ａ及び図３７Ｂは、図３７Ａに示す構成において、リボークデバイスを含む下位エンティティから更新サブＥＫＢを受信した上位エンティティにおいて生成するＥＫＢの例を説明する図である。リボークデバイスを含む下位エンティティの頂点ノードは、上位エンティティの末端ノード（ｎｏｄｅ１００）３６０１に対応する。

上位エンティティは、上位エンティティのサブルートから末端ノード（ｎｏｄｅ１００）３６０１までのパスに存在するノードキーを更新して新たな更新サブＥＫＢを生成する。更新サブＥＫＢは、図３７Ｂのようになる。更新されたキーは、下線及び［’］を付して示してある。このように更新された末端ノードからサブルートまでのパス上のノードキーを更新してそのエンティティにおける更新サブＥＫＢとする。

次に、リボークする対象をエンティティとした場合の処理、すなわちエンティティのリボーク処理について説明する。

図３８Ａは、エンティティ管理によるキー配信ツリー構造を示している。ツリー最上位にはルートノードが設定され、その数段下にエンティティＡ０１～Ａｎｎ、さらにその下位段にＢ０１～Ｂｎｋのエンティティ、さらにその下位段にＣ１～ｃｎのエンティティが構成されている。最も下のエンティティは、末端ノード（リーフ）が個々のデバイス、例えば記録再生器、再生専用器等であるとする。

ここで、リボーク処理を、エンティティＣｎ，３７３０に対して実行する場合について説明する。最下段のエンティティＣｎ，３７３０は、図３８Ｂに示すよ

うに頂点ノード３４３１を持ち、末端ノードであるリーフに複数のデバイスを持つ構成である。

エンティティＣｎ，３７３０をリボークすることにより、エンティティＣｎ，３７３０に属する全てのデバイスのツリー構造からの一括排除が可能となる。エンティティＣｎ，３７３０のリボーク処理は、エンティティＣｎ，３７３０の上位エンティティであるエンティティＢｎｋ，３７２０において実行される。エンティティＢｎｋ，３７２０は、エンティティＣｎ，３７３０の頂点ノード３７３１を末端ノードとして有するエンティティである。

エンティティＢｎｋ，３７２０は、下位エンティティＣｎ，３７３０のリボークを実行する場合、エンティティＣｎｋ，３７３０の頂点ノード３７３１に対応するエンティティＢｎｋ，３７２０の末端ノード３７３１を更新し、さらに、そのリボークエンティティ３７３０からエンティティＢｎｋ，３７２０のサブルートまでのパス上のノードキーの更新を行ない有効化キーブロックを生成して更新サブＥＫＢを生成する。更新対象となるノードキーは、図３８Ｃのサブルート３７２１からリボークエンティティの頂点ノードを構成する末端ノード３７３１に至るパス上のノードキーである。すなわち、ノード３７２１，３７２４，３７２５，３７３１のノードキーが更新対象となる。これら各ノードのノードキーを更新してエンティティＢｎｋ，３７２０の新たな更新サブＥＫＢを生成する。

あるいは、エンティティＢｎｋ，３７２０は、下位エンティティＣｎ，３７３０のリボークを実行する場合、エンティティＣｎｋ，３７３０の頂点ノード３７３１に対応するエンティティＢｎｋ，３７２０の末端ノード３７３１は更新せず、そのリボークエンティティ３７３０からエンティティＢｎｋ，３７２０のサブルートまでのパス上の末端ノード３７３１を除くノードキーの更新を行ない有効化キーブロックを生成して更新サブＥＫＢを生成してもよい。

さらに、エンティティＢｎｋ，３７２０が更新した有効化キーブロック（サブＥＫＢ）は、上位エンティティに送信される。この場合、上位エンティティはエンティティＡｎｎ，３７１０であり、エンティティＢｎｋ，３７２０の頂点ノード３７２１を末端ノードとして有するエンティティである。

エンティティＡｎｎ，３７１０は、下位エンティティＢｎｋ，３７２０から有

効化キーブロック（サブＥＫＢ）を受領すると、そのキーブロックに含まれるエンティティＢｎｋ，３７２０の頂点ノード３７２１に対応するエンティティＡｎｎ，３７１０の末端ノード３７２１を、下位エンティティＢｎｋ，３７２０において更新されたキーに設定して、自身のエンティティＡｎｎ，３７１０のサブＥＫＢの更新処理を実行する。図３８Ｄにエンティティ Ａｎｎ，３７１０のツリー構成を示す。エンティティＡｎｎ，３７１０において、更新対象となるノードキーは、図３８Ｄのサブルート３７１１から更新サブＥＫＢを送信してきたエンティティのノード３７２１に連なるパスを構成する各ノード３７１１，３７１４，３７１５のノードキーである。これら各ノードのノードキーを更新してエンティティＡｎｎ，３７１０の新たな更新サブＥＫＢを生成する。

これらの処理を順次、上位のエンティティにおいて実行し、図３０Ｂで説明したルートエンティティまで実行する。この一連の処理により、エンティティのリボーク処理が完結する。なお、それぞれのエンティティにおいて更新されたサブＥＫＢは、最終的にキー発行センタ（ＫＤＣ）に送信され、保管される。キー発行センタ（ＫＤＣ）は、全てのエンティティの更新サブＥＫＢに基づいて、様々なＥＫＢを生成する。更新ＥＫＢは、リボークされたエンティティに属するデバイスでの復号が不可能な暗号化キーブロックとなる。

エンティティのリボーク処理のシーケンス図を図３９に示す。処理手順を図３９のシーケンス図に従って説明する。まず、エンティティをリボークしようとするエンティティ管理エンティティ（Ｅ－Ｅｎ）は、エンティティ管理エンティティ（Ｅ－Ｅｎ）内のリボーク対象の末端ノードを排除するために必要なキー更新を行ない、エンティティ管理エンティティ（Ｅ－Ｅｎ）の新たなサブＥＫＢ（Ｅ）を生成する。更新サブＥＫＢ（Ｅ）は、上位エンティティに送付される。更新サブＥＫＢ（Ｅ）を受領した上位（親）エンティティ（Ｐ１－Ｅｎ）は、更新サブＥＫＢ（Ｅ）の更新頂点ノードに対応した末端ノードキーの更新及び、その末端ノードからサブルートに至るパス上のノードキーを更新した更新サブＥＫＢ（Ｐ１）を生成する。これらの処理を順次、上位エンティティにおいて実行して、最終的に更新された全てのサブＥＫＢがキー発行センタ（ＫＤＣ）に格納され管理される。キー発行センタ（ＫＤＣ）は、全てのエンティティの更新サブＥ

ＫＢに基づいて、様々なＥＫＢを生成する。更新ＥＫＢは、リボークされたエンティティに属するデバイスでの復号が不可能な暗号化キーブロックとなる。

図４０にリボークされた下位エンティティと、リボークを行なった上位エンティティの対応を説明する図を示す。上位エンティティの末端ノード３９０１は、エンティティのリボークにより更新され、上位エンティティのツリーにおける末端ノード３９０１からサブルートまでのパスに存在するノードキーの更新により、新たなサブＥＫＢが生成される。その結果、リボークされた下位エンティティの頂点ノード３９０２のノードキーと、上位エンティティの末端ノード３９０１のノードキーは不一致となる。エンティティのリボーク後にキー発行センタ（ＫＤＣ）によって生成されるＥＫＢは、上位エンティティにおいて更新された末端ノード３９０１のキーに基づいて生成されることになるので、その更新キーを保有しない下位エンティティのリーフに対応するデバイスは、キー発行センタ（ＫＤＣ）によって生成されるＥＫＢの復号ガ不可能になる。

なお、上述の説明では、デバイスを管理する最下段のエンティティのリボーク処理について説明したが、ツリーの中段にあるエンティティ管理エンティティをその上位エンティティがリボークする処理も上記と同様のプロセスによって可能である。中段のエンティティ管理エンティティをリボークすることにより、リボークされたエンティティ管理エンティティの下位に属する全ての複数エンティティ及びデバイスを一括してリボーク可能となる。

このように、エンティティ単位でのリボークを実行することにより、１つ１つのデバイス単位で実行するリボーク処理に比較して簡易なプロセスでのリボーク処理が可能となる。

［エンティティのケイパビリティ管理］

次に、エンティティ単位でのキー配信ツリー構成において、各エンティティの許容するケイパビリティ(Capability)を管理して、ケイパビリティに応じたコンテンツ配信を行う処理構成について説明する。ここでケイパビリティとは、例えば特定の圧縮音声データの復号が可能であるとか、特定の音声再生方式を許容するとか、あるいは特定の画像処理プログラムを処理できる等、デバイスがどのようなコンテンツ、あるいはプログラム等を処理できるデバイスであるか、すなわ

ちデバイスのデータ処理能力の定義情報である。

図４１にケイパビリティを定義したエンティティ構成例を示す。キー配信ツリー構成の最頂点ににルートノードが位置し、下層に複数のエンティティが接続されて各ノードが２分岐を持つツリー構成である。ここで、例えばエンティティ４００１は、音声再生方式Ａ，Ｂ，Ｃのいずれかを許容するケイパビリティを持つエンティティとして定義される。具体的には、例えばある音声圧縮プログラムーＡ、Ｂ、又はＣ方式で圧縮した音楽データを配信した場合に、エンティティ４００１以下に構成されたエンティティに属するデバイスは圧縮データを伸長する処理が可能である。

同様にエンティティ４００２は音声再生方式Ｂ又はＣ、エンティティ４００３は音声再生方式Ａ又はＢ、エンティティ４００４は音声再生方式Ｂ、エンティティ４００５は音声再生方式Ｃを処理することが可能なケイパビリティを持つエンティティとして定義される。

一方、エンティティ４０２１は、画像再生方式ｐ，ｑ，ｒを許容するエンティティとして定義され、エンティティ４０２２は方式ｐ，ｑの画像再生方式、エンティティ４０２３は方式ｐの画像再生が可能なケイパビリティを持つエンティティとして定義される。

このような各エンティティのケイパビリティ情報は、キー発行センタ（ＫＤＣ）において管理される。キー発行センタ（ＫＤＣ）は、例えばあるコンテンツプロバイダが特定の圧縮プログラムで圧縮した音楽データを様々なデバイスに配信したい場合、その特定の圧縮プログラムを再生可能なデバイスに対してのみ復号可能な有効化キーブロック（ＥＫＢ）を各エンティティのケイパビリティ情報に基づいて生成することができる。コンテンツを提供するコンテンツプロバイダは、ケイパビリティ情報に基づいて生成した有効化キーブロック（ＥＫＢ）によって暗号化したコンテンツキーを配信し、そのコンテンツキーで暗号化した圧縮音声データを各デバイスに提供する。この構成により、データの処理が可能なデバイスに対してのみ特定の処理プログラムを確実に提供することが可能となる。

なお、図４１では全てのエンティティについてケイパビリティ情報を定義している構成であるが、図４１の構成ように全てのエンティティにケイパビリティ情

報を定義することは必ずしも必要ではなく、例えば図４２に示すようにデバイスが属する最下段のエンティティについてのみケイパビリティを定義して、最下段のエンティティに属するデバイスのケイパビリティをキー発行センタ（ＫＤＣ）において管理して、コンテンツプロバイダが望む処理の可能なデバイスにのみ復号可能な有効化キーブロック（ＥＫＢ）を最下段のエンティティに定義されたケイパビリティ情報に基づいて生成する構成としてもよい。図４２では、末端ノードにデバイスが定義されたエンティティ４１０１＝４１０５におけるケイパビリティが定義され、これらのエンティティについてのケイパビリティをキー発行センタ（ＫＤＣ）において管理する構成である。例えばエンティティ４１０１には音声再生については方式Ｂ、画像再生については方式ｒの処理が可能なデバイスが属している。エンティティ４１０２には音声再生については方式Ａ、画像再生については方式ｑの処理が可能なデバイスが属している等である。

図４３Ａ及び図４３Ｂにキー発行センタ（ＫＤＣ）において管理するケイパビリティ管理テーブルの構成例を示す。ケイパビリティ管理テーブルは、図４３Ａのようなデータ構成を持つ。すなわち、各エンティティを識別する識別子としてのエンティティＩＤ、そのエンティティに定義されたケイパビリティを示すケイパビリティリスト、このケイパビリティリストは図４３Ｂに示すように、例えば音声データ再生処理方式（Ａ）が処理可能であれば［１］、処理不可能であれは［０］、音声データ再生処理方式（Ｂ）が処理可能であれば［１］、処理不可能であれは［０］…等、様々な態様のデータ処理についての可否を１ビットづつ［１］又は［０］を設定して構成されている。なお、このケイパビリティ情報の設定方法はこのような形式に限らず、エンティティの管理デバイスについてのケイパビリティを識別可能であれば他の構成でもよい。

ケイパビリティ管理テーブルには、さらに、各エンティティのサブＥＫＢ、あるいはサブＥＫＢが別のデータベースに格納されている場合は、サブＥＫＢの識別情報が格納され、さらに、各エンティティのサブルートノード識別データが格納される。

キー発行センタ（ＫＤＣ）は、ケイパビリティ管理テーブルに基づいて、例えば特定のコンテンツの再生可能なデバイスのみが復号可能な有効化キーブロック

（ＥＫＢ）を生成する。図４４を用いて、ケイパビリティ情報に基づく有効化キーブロックの生成処理について説明する。

まず、ステップＳ４３０１において、キー発行センタ（ＫＤＣ）は、ケイパビリティ管理テーブルから、指定されたケイパビリティを持つエンティティを選択する。具体的には、例えばコンテンツプロバイダが音声データ再生処理方式Ａに基づく再生可能なデータを配信したい場合は、図４３Ａのケイパビリティリストから、例えば音声データ再生処理（方式Ａ）の項目が［１］に設定されたエンティティを選択する。

次に、ステップＳ４３０２において、選択されたエンティティによって構成される選択エンティティＩＤのリストを生成する。次に、ステップＳ４３０３で、選択エンティティＩＤによって構成されるツリーに必要なパス（キー配信ツリー構成のパス）を選択する。ステップＳ４３０４では、選択エンティティＩＤのリストに含まれる全てのパス選択が完了したか否かを判定し、完了するまで、ステップＳ４３０３においてパスを生成する。これは、複数のエンティティが選択された場合に、それぞれのパスを順次選択する処理を意味している。

選択エンティティＩＤのリストに含まれる全てのパス選択が完了すると、ステップＳ４３０５に進み、選択したパスと、選択エンティティによってのみ構成されるキー配信ツリー構造を構築する。

次に、ステップＳ４３０６において、ステップＳ４３０５で生成したツリー構造のノードキーの更新処理を行ない、更新ノードキーを生成する。さらに、ツリーを構成する選択エンティティのサブＥＫＢをケイパビリティ管理テーブルから取り出し、サブＥＫＢと、ステップＳ４３０６で生成した更新ノードキーとに基づいて選択エンティティのデバイスにおいてのみ復号可能な有効化キーブロック（ＥＫＢ）を生成する。このようにして生成した有効化キーブロック（ＥＫＢ）は、特定のケイパビリティを持つデバイスにおいてのみ利用、すなわち復号可能な有効化キーブロック（ＥＫＢ）となる。この有効化キーブロック（ＥＫＢ）で例えばコンテンツキーを暗号化して、そのコンテンツキーで特定プログラムに基づいて圧縮したコンテンツを暗号化してデバイスに提供することで、キー発行センタ（ＫＤＣ）によって選択された特定の処理可能なデバイスにおいてのみコン

テンツが利用される。

このようにキー発行センタ（ＫＤＣ）は、ケイパビリティ管理テーブルに基づいて、例えば特定のコンテンツの再生可能なデバイスのみが復号可能な有効化キーブロック（ＥＫＢ）を生成する。従って、新たなエンティティが登録される場合には、その新規登録エンティティのケイパビリティを予め取得することが必要となる。このエンティティ新規登録に伴うケイパビリティ通知処理について図４５を用いて説明する。

図４５は、新規エンティティがキー配信ツリー構成に参加する場合のケイパビリティ通知処理シーケンスを示した図である。

新たにツリー構成中に追加される新規（子）エンティティ（Ｎ－Ｅｎ）は、上位（親）エンティティ（Ｐ－Ｅｎ）に対して新規登録要求を実行する。なお、各エンティティは、公開鍵暗号方式に従った公開鍵を保有し、新規エンティティは自己の公開鍵を登録要求に際して上位エンティティ（Ｐ－Ｅｎ）に送付する。

登録要求を受領した上位エンティティ（Ｐ－Ｅｎ）は、受領した新規（子）エンティティ（Ｎ－Ｅｎ）の公開鍵を証明書発行局（ＣＡ：Certificate Authority）に転送し、ＣＡの署名を付加した新規（子）エンティティ（Ｎ－Ｅｎ）の公開鍵を受領する。これらの手続きは、上位エンティティ（Ｐ－Ｅｎ）と新規（子）エンティティ（Ｎ－Ｅｎ）との相互認証の手続きとして行われる。

これらの処理により、新規登録要求エンティティの認証が終了すると、上位エンティティ（Ｐ－Ｅｎ）は、新規（子）エンティティ（Ｎ－Ｅｎ）の登録を許可し、新規（子）エンティティ（Ｎ－Ｅｎ）のノードキーを新規（子）エンティティ（Ｎ－Ｅｎ）に送信する。このノードキーは、上位エンティティ（Ｐ－Ｅｎ）の末端ノードの１つのノードキーであり、かつ、新規（子）エンティティ（Ｎ－Ｅｎ）の頂点ノード、すなわちサブルートキーに対応する。

このノードキー送信が終了すると、新規（子）エンティティ（Ｎ－Ｅｎ）は、新規（子）エンティティ（Ｎ－Ｅｎ）のツリー構成を構築し、構築したツリーの頂点に受信した頂点ノードのサブルートキーを設定し、各ノード、リーフのキーを設定して、エンティティ内の有効化キーブロック（サブＥＫＢ）を生成する。一方、上位エンティティ（Ｐ－Ｅｎ）も、新規（子）エンティティ（Ｎ－Ｅｎ）

の追加により、有効化する末端ノードを追加した上位エンティティ（Ｐ－Ｅｎ）内のサブＥＫＢを生成する。

新規（子）エンティティ（Ｎ－Ｅｎ）は、新規（子）エンティティ（Ｎ－Ｅｎ）内のノードキー、リーフキーによって構成されるサブＥＫＢを生成すると、これを上位エンティティ（Ｐ－Ｅｎ）に送信し、さらに、自己のエンティティで管理するデバイスについてのケイパビリティ情報を上位エンティティに通知する。

新規（子）エンティティ（Ｎ－Ｅｎ）からサブＥＫＢ及びケイパビリティ情報を受信した上位エンティティ（Ｐ－Ｅｎ）は、受信したサブＥＫＢとケイパビリティ情報と、上位エンティティ（Ｐ－Ｅｎ）の更新したサブＥＫＢとをキー発行センタ（ＫＤＣ：Key Distribute Center）に送信する。

キー発行センタ（ＫＤＣ）は、受領したエンティティのサブＥＫＢ及びケイパビリティ情報とを図４３Ａ及び図４３Ｂで説明したケイパビリティ管理テーブルに登録し、ケイパビリティ管理テーブルを更新する。キー発行センタ（ＫＤＣ）は、更新したケイパビリティ管理テーブルに基づいて、様々な態様のＥＫＢ、すなわち特定のケイパビリティを持つエンティティあるいはデバイスのみが復号可能なＥＫＢを生成することが可能となる。

以上、特定の実施例を参照しながら、本発明について詳解してきた。しかしながら、本発明の要旨を逸脱しない範囲で当業者が実施例の修正や代用を成し得ることは自明である。すなわち、例示という形態で本発明を開示してきたのであり、限定的に解釈されるべきではない。本発明の要旨を判断するためには、冒頭に記載した特許請求の範囲の欄を参酌すべきである。

産業上の利用可能性

以上、説明したように、本発明の情報処理システム及び方法によれば、コンテンツキーや認証キー、コンテンツチェック値生成キー、プログラムデータ等の暗号化処理キーブロックとして適用可能な有効化キーブロック（ＥＫＢ）の生成において、階層的鍵配信ツリーを配信デバイスに応じて再構築して、再構築した簡略ツリーに含まれるノード、リーフに基づいて有効化キーブロック（ＥＫＢ）を

生成する構成としたので、有効化キーブロック（ＥＫＢ）の大幅なデータ量削減が実現される。

また、本発明の情報処理システム及び方法によれば、簡略化した再構築階層ツリーに基づく有効化キーブロック（ＥＫＢ）を構成し、さらに、ＥＫＢ中の暗号化キーデータの位置識別子としてのタグに暗号化キーデータの有無を判別するデータを含ませた構成としたので、ＥＫＢの大幅なデータ量削減が実現されるとともに、ＥＫＢを受領したデバイスでのタグを用いた暗号化キーデータの抽出が容易となり、デバイスでのＥＫＢ復号処理が効率的になる。

請求の範囲

１．１以上の選択されたデバイスにおいてのみ利用可能な暗号化メッセージデータを配信する情報処理システムであり、

個々のデバイスは、複数の異なるデバイスをリーフとした階層ツリー構造における各ノードに固有のノードキーと各デバイス固有のリーフキーの異なるキーセットをそれぞれ保有するとともに、デバイスに対して配信される前記暗号化メッセージデータについての復号処理を前記キーセットを使用して実行する暗号処理手段を有し、

前記デバイスに提供される暗号化メッセージデータは、前記階層ツリー構造の１つのノードを頂点ノードとし、該頂点ノードの下位に連結されるノード及びリーフによって構成されるグループ内のノードキーの少なくともいずれかを更新した更新ノードキーを、該グループのノードキーあるいはリーフキーによって暗号化した暗号化キーデータを含む有効化キーブロック（ＥＫＢ）の復号処理によって得られる前記更新ノードキーによって暗号化されたデータ構成であり、

前記有効化キーブロック（ＥＫＢ）は、前記暗号化キーデータによって構成されるデータ部と、該データ部に格納される暗号化キーデータの前記階層ツリー構造における位置識別データとしてのタグ部とを含む構成であることを特徴とする情報処理システム。

２．前記有効化キーブロック（ＥＫＢ）に含まれる前記暗号化キーデータは、前記階層ツリー構造を構成するノードキーを下位ノードキー又は下位リーフキーを用いて暗号化したデータであり、

前記タグ部に格納される位置識別データは、前記有効化キーブロック（ＥＫＢ）に格納された１以上の暗号化キーデータ各々のノード位置の下位の左右ノード又はリーフ位置の暗号化キーデータの有無を示すタグとして構成されていることを特徴とする請求の範囲第１項に記載の情報処理システム。

３．前記有効化キーブロック（ＥＫＢ）に含まれる前記暗号化キーデータは、該有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段とした簡略化した２分岐型ツリーを構成するパスを選択して不要ノードを省略するこ

とにより再構築される再構築階層ツリーのノード又はリーフに対応するキーのみに基づいて構成され、

前記タグ部に格納される位置識別データは、前記有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示すデータを含む構成であることを特徴とする請求の範囲第１項に記載の情報処理システム。

４．前記有効化キーブロック（ＥＫＢ）に含まれる前記暗号化キーデータは、該有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段とした簡略化した２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーのノード又はリーフに対応するキーのみに基づいて構成され、

前記タグ部に格納される位置識別データは、前記有効化キーブロック（ＥＫＢ）に格納された１以上の暗号化キーデータ各々のノード位置の下位の左右ノード又はリーフ位置の暗号化キーデータの有無を示すタグと、該タグに対応する暗号化キーの格納の有無を示すデータを含む構成であることを特徴とする請求の範囲第１項に記載の情報処理システム。

５．前記再構築階層ツリーは、共通要素を持つデバイスの部分集合ツリーとして定義されるエンティティの頂点ノードであるサブルートを選択して構成されるツリーであることを特徴とする請求の範囲第４項に記載の情報処理システム。

６．前記有効化キーブロック（ＥＫＢ）に含まれる前記暗号化キーデータは、該有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段とした簡略化した多分岐型ツリーにおいて、前記末端ノード又はリーフと、該多分岐型ツリーの頂点とを直接接続するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーの頂点ノード及び末端ノード又はリーフに対応するキーのみに基づいて構成され、

前記タグ部に格納される位置識別データは、前記有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示すデータを含む構成であることを特徴とする請求の範囲第１項に記載の情報処理システム。

７．前記再構築階層ツリーは、簡略化した多分岐型ツリーを構成する頂点ノードと、簡略化したツリーを構成する末端ノード又はリーフとを直接、接続した３以

上の分岐を持つツリーであることを特徴とする請求の範囲第６項に記載の情報処理システム。

８．前記デバイスにおける前記暗号処理手段は、前記有効化キーブロック（ＥＫＢ）の前記タグ部のデータにより、前記暗号化キーデータを順次抽出して、復号処理を実行し、前記更新ノードキーを取得し、該取得した更新ノードキーにより前記暗号化メッセージデータの復号を実行する構成であることを特徴とする請求の範囲第１項に記載の情報処理システム。

９．前記メッセージデータは、コンテンツデータを復号するための復号鍵として使用可能なコンテンツキーであることを特徴とする請求の範囲第１項に記載の情報処理システム。

１０．前記メッセージデータは、認証処理において用いられる認証キーであることを特徴とする請求の範囲第１項に記載の情報処理システム。

１１．前記メッセージデータは、コンテンツのインテグリティ・チェック値（ＩＣＶ）生成キーであることを特徴とする請求の範囲第１項に記載の情報処理システム。

１２．前記メッセージデータは、プログラムコードであることを特徴とする請求の範囲第１項に記載の情報処理システム。

１３．１以上の選択されたデバイスにおいてのみ利用可能な暗号化メッセージデータを配信する情報処理方法であり、

複数の異なるデバイスをリーフとした階層ツリー構造の１つのノードを頂点ノードとし、該頂点ノードの下位に連結されるノード及びリーフによって構成されるグループ内のノードキーの少なくともいずれかを更新した更新ノードキーを、該グループのノードキーあるいはリーフキーによって暗号化した暗号化キーデータを含むデータ部と、該データ部に格納される暗号化キーデータの前記階層ツリー構造における位置識別データとしてのタグ部とを含む有効化キーブロック（ＥＫＢ）を生成する有効化キーブロック（ＥＫＢ）生成ステップと、

前記更新ノードキーによって暗号化したメッセージデータを生成してデバイスに対して配信するメッセージデータ配信ステップとを有する情報処理方法。

１４．前記情報処理方法は、さらに、

前記階層ツリー構造における各ノードに固有のノードキーと各デバイス固有のリーフキーの異なるキーセットをそれぞれ保有するデバイスにおいて、前記暗号化メッセージデータについての復号処理を前記キーセットを使用して実行する復号処理ステップを有する請求の範囲第１３項に記載の情報処理方法。

１５．前記有効化キーブロック（ＥＫＢ）生成ステップは、

前記階層ツリー構造を構成するノードキーを下位ノードキー又は下位リーフキーを用いて暗号化して前記暗号化キーデータを生成するステップと、

前記有効化キーブロック（ＥＫＢ）に格納される１以上の暗号化キーデータ各々のノード位置の下位の左右位置のノード又はリーフ位置の暗号化キーデータの有無を示すタグを生成して前記タグ部に格納するステップとを含むことを特徴とする請求の範囲第１３項に記載の情報処理方法。

１６．前記有効化キーブロック（ＥＫＢ）生成ステップは、

該有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段とした簡略化した２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築階層ツリーを生成するステップと、

前記再構築階層ツリーの構成ノード又はリーフに対応するキーのみに基づいて有効化キーブロック（ＥＫＢ）を生成するステップと、

前記有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示すデータを前記タグ部に格納するステップとを含むことを特徴とする請求の範囲第１３項に記載の情報処理方法。

１７．前記再構築階層ツリーを生成するステップは、共通要素を持つデバイスの部分集合ツリーとして定義されるエンティティの頂点ノードであるサブルートを選択して実行されるツリー生成処理であることを特徴とする請求の範囲第１６項に記載の情報処理方法。

１８．前記有効化キーブロック（ＥＫＢ）生成ステップは、

該有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段とした簡略化した多分岐型ツリーにおいて、前記末端ノード又はリーフと、該多分岐型ツリーの頂点とを直接接続するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーを生成するステップと、

前記有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示すデータを前記タグ部に格納するステップとを含むことを特徴とする請求の範囲第１３項に記載の情報処理方法。

１９．前記再構築階層ツリーの生成ステップにおいて生成する再構築階層ツリーは、簡略化した多分岐型ツリーを構成する頂点ノードと、簡略化したツリーを構成する末端ノード又はリーフとを直接、接続した３以上の分岐を持つツリーとして生成することを特徴とする請求の範囲第１８項に記載の情報処理方法。

２０．前記復号処理ステップは、

前記有効化キーブロック（ＥＫＢ）の前記タグ部に格納された位置識別データに基づいて前記データ部に格納された暗号化キーデータを順次抽出して順次復号処理を実行することにより前記更新ノードキーを取得する更新ノードキー取得ステップと、

前記更新ノードキーにより前記暗号化メッセージデータの復号を実行するメッセージデータ復号ステップとを含むことを特徴とする請求の範囲第１４項に記載の情報処理方法。

２１．前記メッセージデータは、コンテンツデータを復号するための復号鍵として使用可能なコンテンツキーであることを特徴とする請求の範囲第１３項に記載の情報処理方法。

２２．前記メッセージデータは、認証処理において用いられる認証キーであることを特徴とする請求の範囲第１３項に記載の情報処理方法。

２３．前記メッセージデータは、コンテンツのインテグリティ・チェック値（ＩＣＶ）生成キーであることを特徴とする請求の範囲第１３項に記載の情報処理方法。

２４．前記メッセージデータは、プログラムコードであることを特徴とする請求の範囲第１３項に記載の情報処理方法。

２５．データを格納した情報記録媒体であり、

複数の異なるデバイスをリーフとした階層ツリー構造の１つのノードを頂点ノードとし、該頂点ノードの下位に連結されるノード及びリーフによって構成されるグループ内のノードキーの少なくともいずれかを更新した更新ノードキーを、

該グループのノードキーあるいはリーフキーによって暗号化した暗号化キーデータによって構成されるデータ部と、該データ部に格納される暗号化キーデータの前記階層ツリー構造における位置識別データとしてのタグ部とを含む有効化キーブロック（ＥＫＢ）と、

前記更新ノードキーによって暗号化したメッセージデータとを格納した情報記録媒体。

２６．前記有効化キーブロック（ＥＫＢ）に含まれる前記暗号化キーデータは、前記階層ツリー構造を構成するノードキーを下位ノードキー又は下位リーフキーを用いて暗号化したデータであり、

前記タグ部に格納される位置識別データは、前記有効化キーブロック（ＥＫＢ）に格納された１以上の暗号化キーデータ各々のノード位置の下位の左右位置のノード又はリーフ位置の暗号化キーデータの有無を示すタグとして構成されていることを特徴とする請求の範囲第２５項に記載の情報記録媒体。

２７．前記有効化キーブロック（ＥＫＢ）に含まれる前記暗号化キーデータは、該有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段とした簡略化した２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーのノード又はリーフに対応するキーのみに基づいて構成され、

前記タグ部に格納される位置識別データは、前記有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示すデータを含む構成であることを特徴とする請求の範囲第２５項に記載の情報記録媒体。

２８．複数の異なるデバイスをリーフとした階層ツリー構造の１つのノードを頂点ノードとし、該頂点ノードの下位に連結されるノード及びリーフによって構成されるグループ内のノードキーの少なくともいずれかを更新した更新ノードキーを、該グループのノードキーあるいはリーフキーによって暗号化した有効化キーブロック（ＥＫＢ）の生成処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを提供するプログラム提供媒体であって、

前記コンピュータ・プログラムは、

該有効化キーブロック（ＥＫＢ）を復号可能な末端ノード又はリーフを最下段

とした簡略化した２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築階層ツリーを生成するステップと、

前記再構築階層ツリーの構成ノード又はリーフに対応するキーのみに基づいて有効化キーブロック（ＥＫＢ）を生成するステップと、

前記有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示すデータを前記タグ部に格納するステップとを含むことを特徴とするプログラム提供媒体。

２９．複数の異なるデバイスをリーフとした階層ツリー構造における固有のノードキーとリーフキーのキーセットを保持する記憶手段と、

配信される暗号化メッセージデータについての復号処理を前記キーセットを使用して実行する復号処理手段を有し、

前記提供される暗号化メッセージデータは、有効化キーブロック（ＥＫＢ）の前記復号処理手段の復号処理によって得られる更新ノードキーによって暗号化されたデータ構成であり、

前記有効化キーブロック（ＥＫＢ）は、

前記階層ツリー構造の１つのノードを頂点ノードとし、該頂点ノードの下位に連結されるノードおよびリーフキーによって構成されるグループ内のノードキーの少なくともいずれかを更新した前記更新ノードキーを、該グループのノードキーあるいはリーフキーによって暗号化した暗号化キーデータによって構成されるデータ部と、

該データ部に格納される暗号化キーデータの前記階層ツリー構造における位置識別データとしてのタグ部とを含む構成であることを特徴とする情報処理装置。

３０．前記有効化キーブロック（ＥＫＢ）に含まれる前記暗号化キーデータは、前記階層ツリー構造を構成するノードキーを下位ノードキーまたは下位リーフキーを用いて暗号化したデータであり、

前記タグ部に格納される位置識別データは、前記有効化キーブロック（ＥＫＢ）に格納された１以上の暗号化キーデータ各々のノード位置の下位の左右ノードまたはリーフ位置の暗号化キーデータの有無を示すタグとして構成されていることを特徴とする請求の範囲第２９項に記載の情報処理装置。

３１．前記有効化キーブロック（ＥＫＢ）に含まれる前記暗号化キーデータは、該有効化キーブロック（ＥＫＢ）を復号可能な末端ノードまたはリーフを最下段とした簡略化した分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーのノードまたはリーフに対応するキーのみに基づいて構成され、

前記タグ部に格納される位置識別データは、前記有効化キーブロック（ＥＫＢ）のタグに対応する暗号化キーの格納の有無を示すデータを含む構成であることを特徴とする請求の範囲第２９項に記載の情報処理装置。

３２．前記有効化キーブロック（ＥＫＢ）に含まれる前記暗号化キーデータは、該有効化キーブロック（ＥＫＢ）を復号可能な末端ノードまたはリーフを最下段とした簡略化した分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーのノードまたはリーフに対応するキーのみに基づいて構成され、

前記タグ部に格納される位置識別データは、前記有効化キーブロック（ＥＫＢ）に格納された１以上の暗号化キーデータ各々のノード位置の下位の左右ノードまたはリーフ位置の暗号化キーデータの有無を示すタグと、該タグに対応する暗号化キーの格納の有無を示すデータを含む構成であることを特徴とする請求の範囲第２９項に記載の情報処理装置。

３３．前記復号処理手段は、

前記有効化キーブロック（ＥＫＢ）の前記タグ部のデータにより、前記暗号化キーデータを順次抽出して、復号処理を実行し、前記更新ノードキーを取得し、該取得した更新ノードキーにより前記暗号化メッセージデータの復号を実行する構成であることを特徴とする請求の範囲第２９項に記載の情報処理装置。

３４．配信される暗号化メッセージデータを復号処理する情報処理方法であって、

階層ツリー構造の１つのノードを頂点ノードとし、該頂点ノードの下位に連結されるノード及びリーフによって構成されるグループ内のノードキーの少なくともいずれかを更新した更新ノードキーを、上記グループのノードキーあるいはリーフキーによって暗号化した暗号化キーデータを含む有効化キーブロック（ＥＫＢ）より、前記暗号化キーデータを取得する暗号化キーデータ取得ステップと、

取得された前記暗号化キーデータを復号することにより上記更新ノードキーを得る更新ノードキー取得ステップとを含み、

前記有効化キーブロック（ＥＫＢ）は、前記暗号化キーデータによって構成されるデータ部と、該データ部に格納される暗号化キーデータの前記階層ツリー構造における位置識別データとしてのタグ部とを含む構成であることを特徴とする情報処理方法。

３５．前記暗号化キーデータ取得ステップでは、

前記有効化キーブロック（ＥＫＢ）の前記タグ部に格納された位置識別データに基づいて前記データ部に格納された暗号化キーデータを順次抽出するようになし、

前記更新ノードキー取得ステップでは、取得された上記暗号化キーデータを順次復号処理を実行することにより前記更新ノードキーを取得するようになすとともに、

前記更新ノードキーにより前記暗号化メッセージデータの復号を実行する復号処理ステップを更に有することを特徴とする請求の範囲第３４項に記載の情報処理方法。

３６．前記復号処理ステップでは、

前記階層ツリー構造における各ノードに固有のノードキーと各デバイス固有のリーフキーの異なるキーセットを保有し、前記暗号化メッセージデータについての復号処理を前記キーセットを使用して実行するようになすことを特徴とする請求の範囲第３５項に記載の情報処理方法。

３７．上記メッセージデータは、コンテンツデータを復号するための復号鍵として使用可能なコンテンツキーであることを特徴とする請求の範囲第３４項に記載の情報処理方法。

３８．上記メッセージデータは、認証処理において用いられる認証キーであることを特徴とする請求の範囲第３４項に記載の情報処理方法。

３９．上記メッセージデータは、コンテンツのインテグリティ・チェック値（ＩＣＶ）生成キーであることを特徴とする請求の範囲第３４項に記載の情報処理方法。

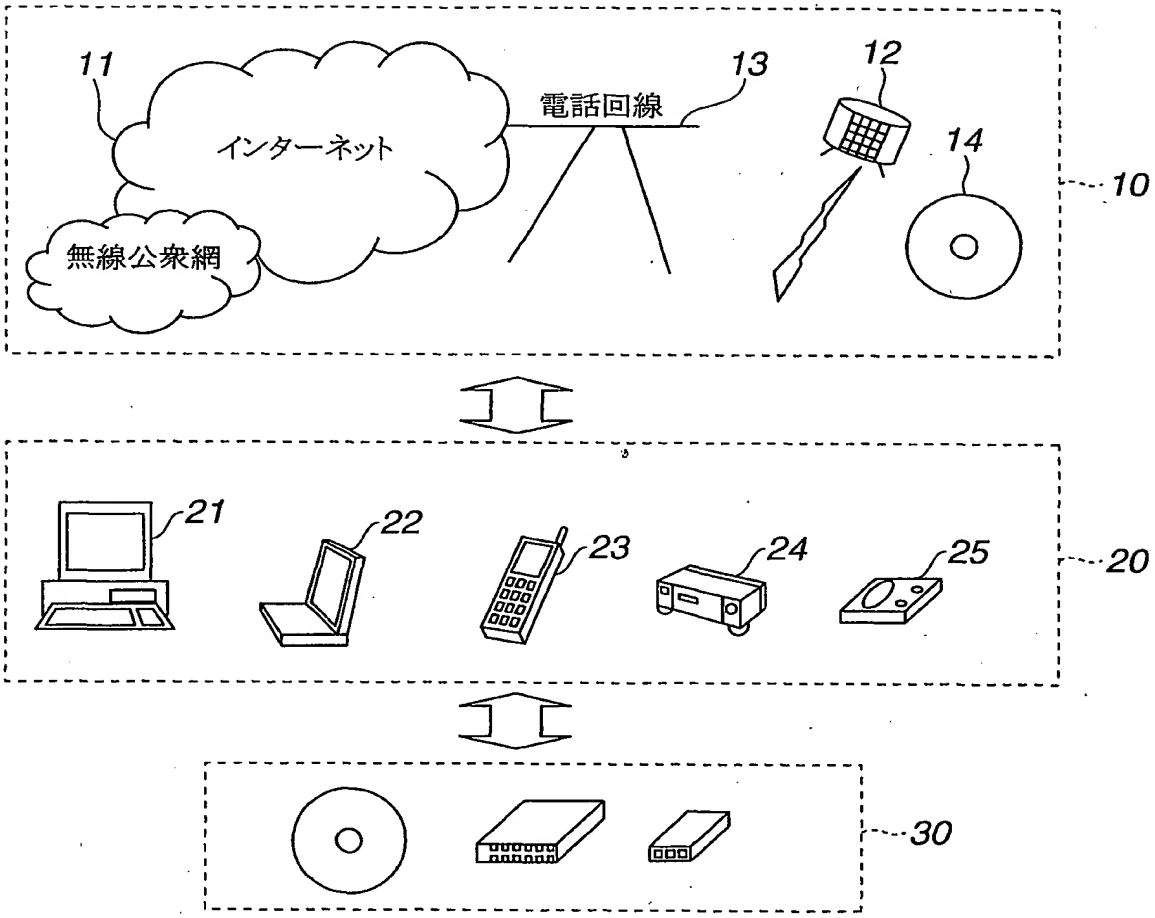


FIG.1

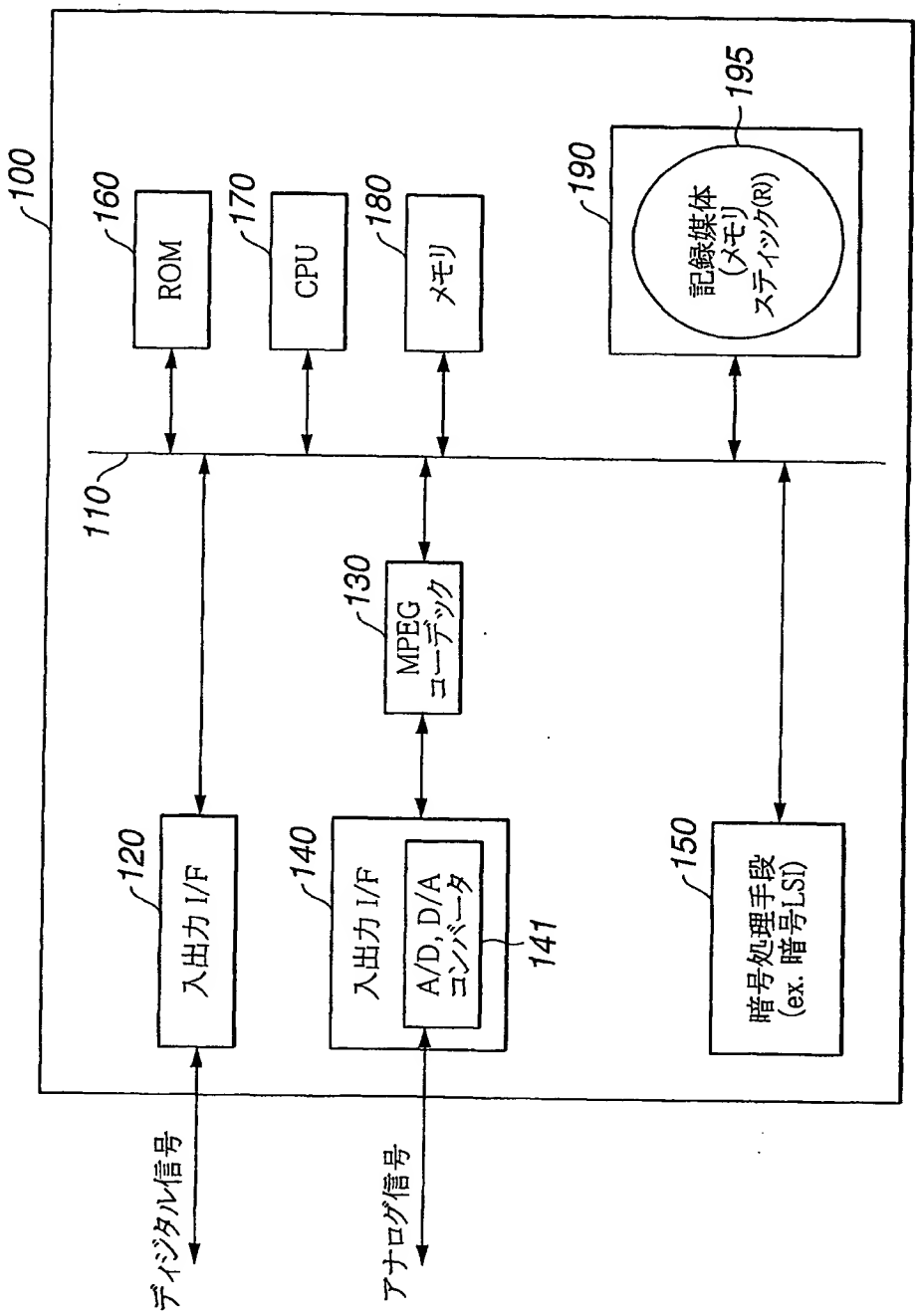

100
160
ROM
170
CPU
180
メモリ
190
記録媒体
(メモリ
スティック(R))
195
110
130
MPEG
コーデック
120
入出力 I/F
140
入出力 I/F
A/D, D/A
コンバータ
141
150
暗号処理手段
(ex. 暗号LSI)
ディジタル信号
アナログ信号

FIG.2

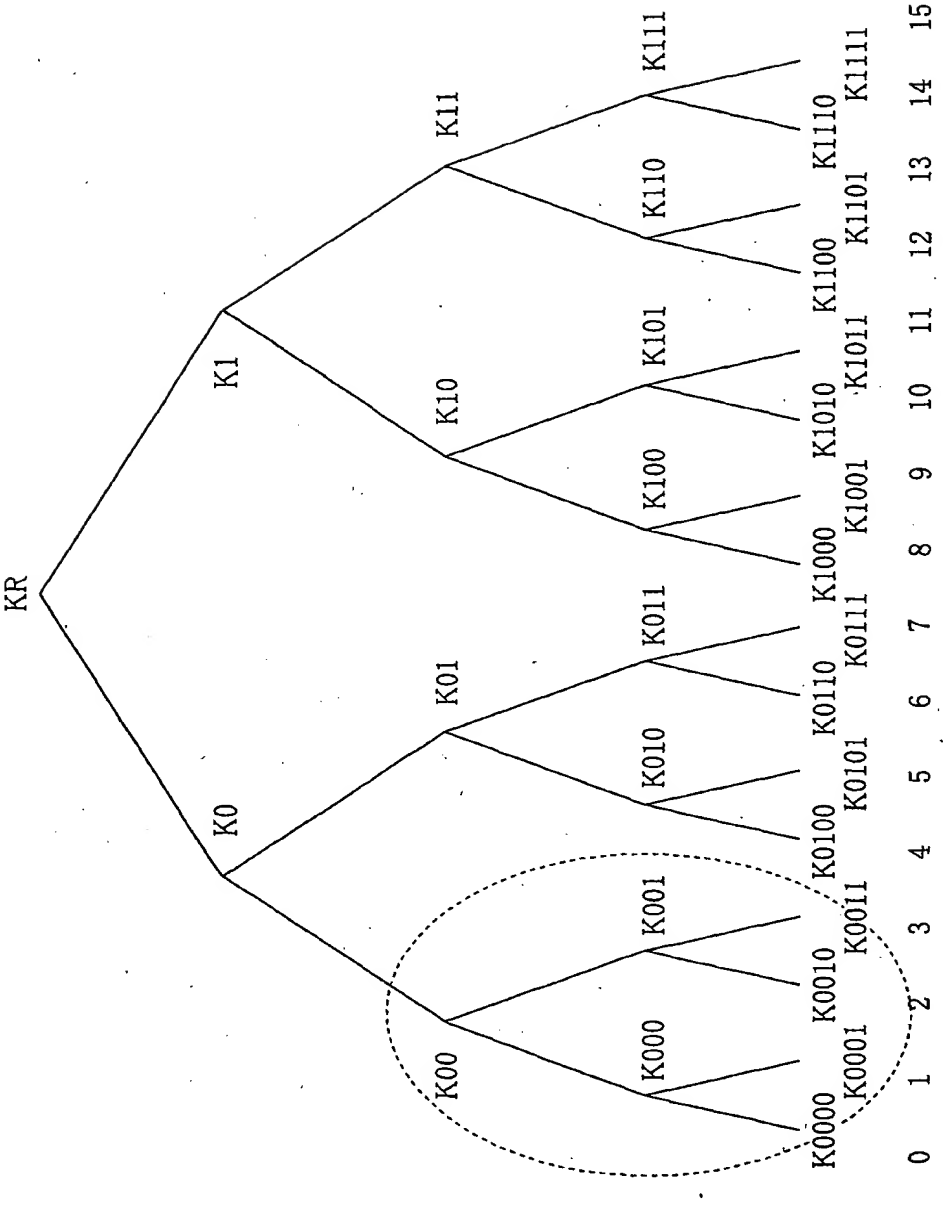

KR
K0
K1
K00
K01
K10
K11
K000
K001
K010
K011
K100
K101
K110
K111
K0000
K0001
K0010
K0011
K0100
K0101
K0110
K0111
K1000
K1001
K1010
K1011
K1100
K1101
K1110
K1111
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
デバイス

FIG.3

バージョン(Version) : t	
インデックス	暗号化キー
0	Enc(K(t)0, K(t)R)
00	Enc(K(t)00, K(t)0)
000	Enc(K000, K(t)00)
001	Enc(K(t)001, K(t)00)
0010	Enc(K0010, K(t)001)

FIG.4A

バージョン(Version) : t	
インデックス	暗号化キー
000	Enc(K000, K(t)00)
001	Enc(K(t)001, K(t)00)
0010	Enc(K0010, K(t)001)

FIG.4B

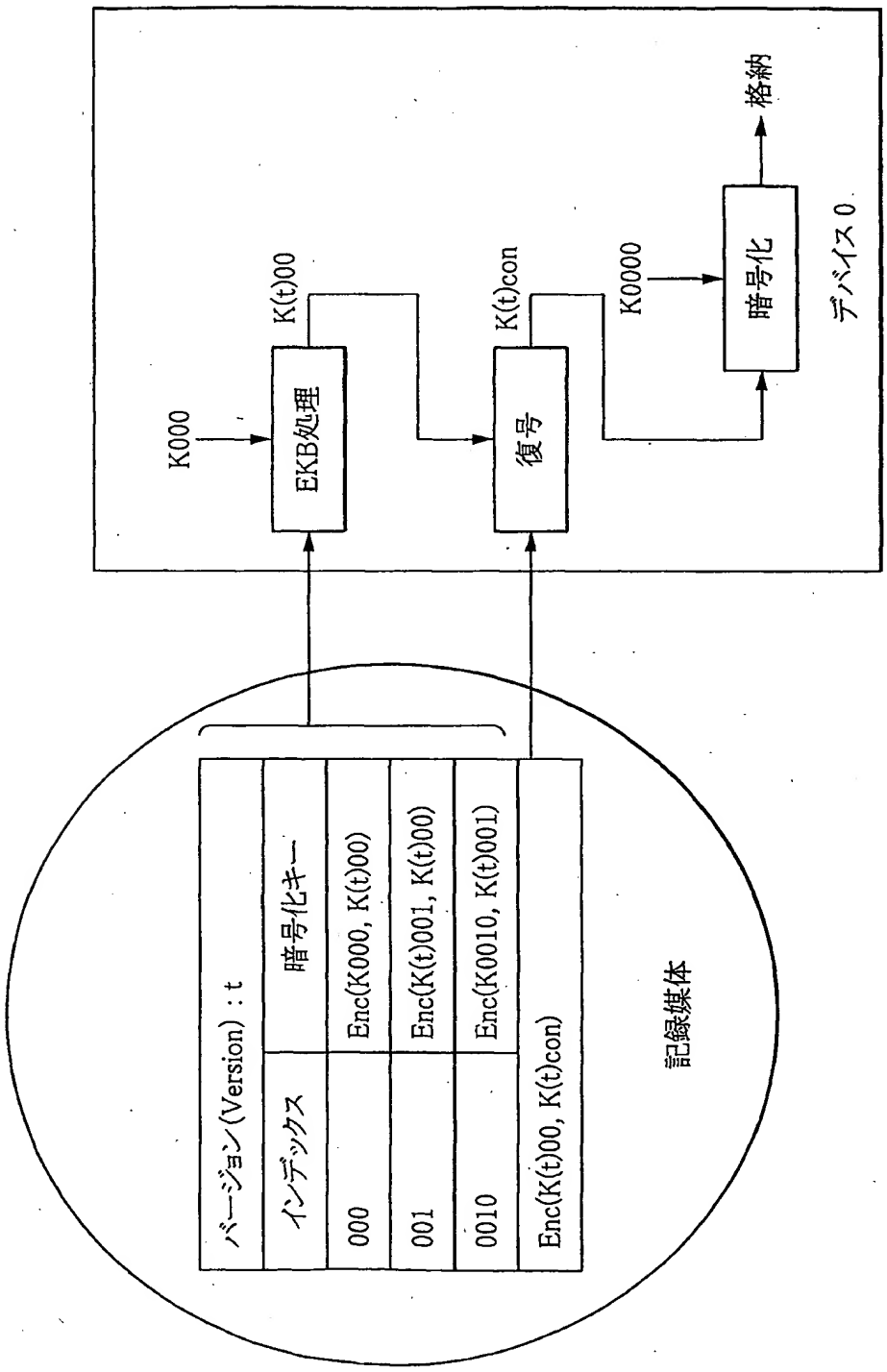

K000
EKB処理
K(t)00
復号
K(t)con
K0000
暗号化
格納
デバイス 0
記録媒体
バージョン(Version) : t
インデックス
暗号化キー
000
Enc(K000, K(t)00)
001
Enc(K(t)001, K(t)00)
0010
Enc(K0010, K(t)001)
Enc(K(t)00, K(t)con)

FIG.5

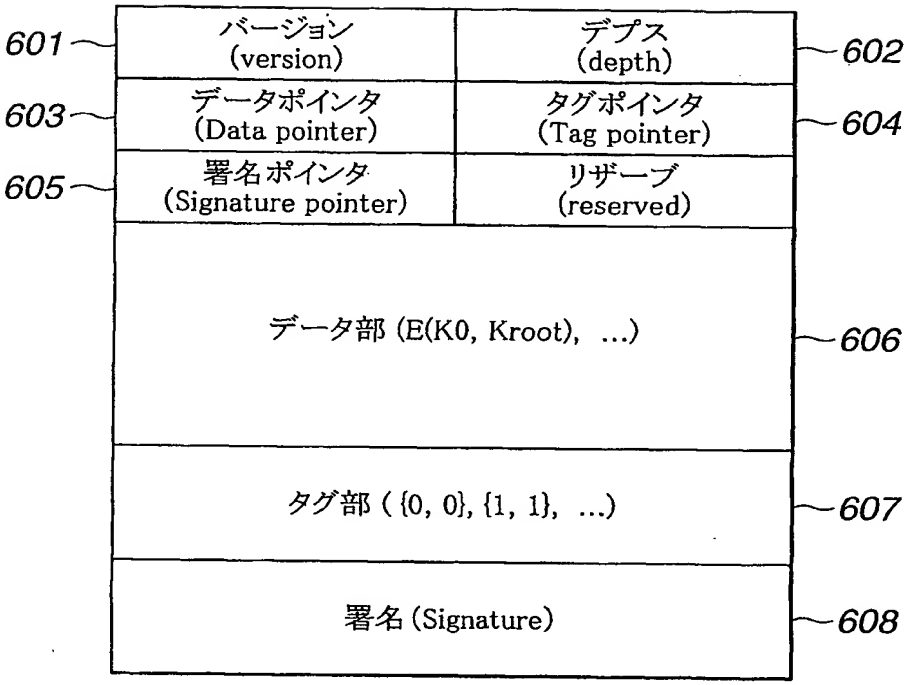

601
バージョン
(version)
デプス
(depth)
602
603
データポインタ
(Data pointer)
タグポインタ
(Tag pointer)
604
605
署名ポインタ
(Signature pointer)
リザーブ
(reserved)
データ部 (E(K0, Kroot), ...)
606
タグ部 ({0, 0}, {1, 1}, ...)
607
署名 (Signature)
608

FIG.6

FIG.7A

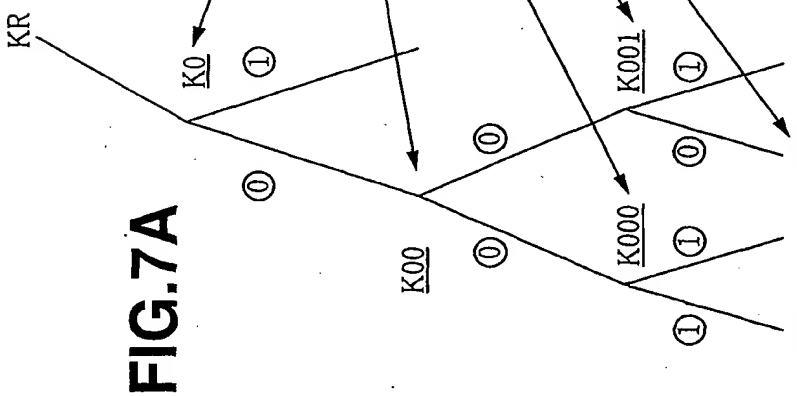


FIG.7B

トップノードアドレス：KR	
データ（暗号化キー）	タグ
Enc(K(t)0, K(t)R)	{0, 1}
Enc(K(t)00, K(t)0)	{0, 0}
Enc(K000, K(t)00)	{1, 1}
Enc(K(t)001, K(t)00)	{0, 1}
Enc(K0010, K(t)001)	{1, 1}

{Lタグ, Rタグ}
左(L) 右(R)それぞれの方向にデータがあれば0、なければ1

FIG.7C

データ：Enc(K(t)0, K(t)R), Enc(K(t)00, K(t)0), ...
タグ：{0, 1}, {0, 0}, {1, 1}, ...

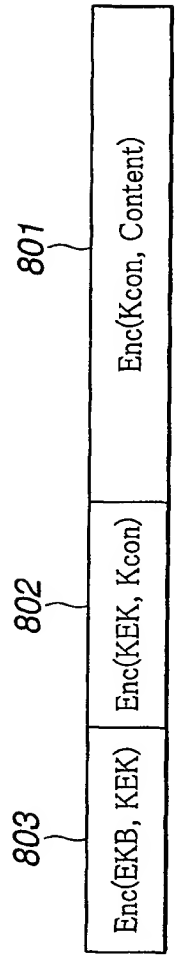

801
802
803
Enc(Kcon, Content)
Enc(KEK, Kcon)
Enc(EKB, KEK)

FIG.8A

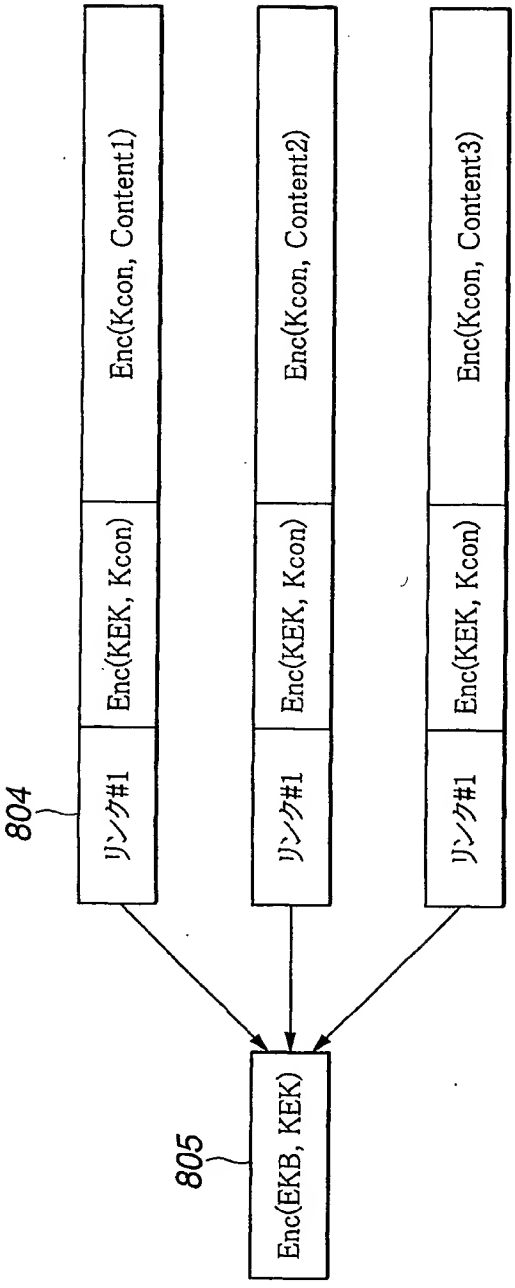

804
805
リンク#1
Enc(KEK, Kcon)
Enc(Kcon, Content1)
リンク#1
Enc(KEK, Kcon)
Enc(Kcon, Content2)
リンク#1
Enc(KEK, Kcon)
Enc(Kcon, Content3)
Enc(EKB, KEK)

FIG.8B

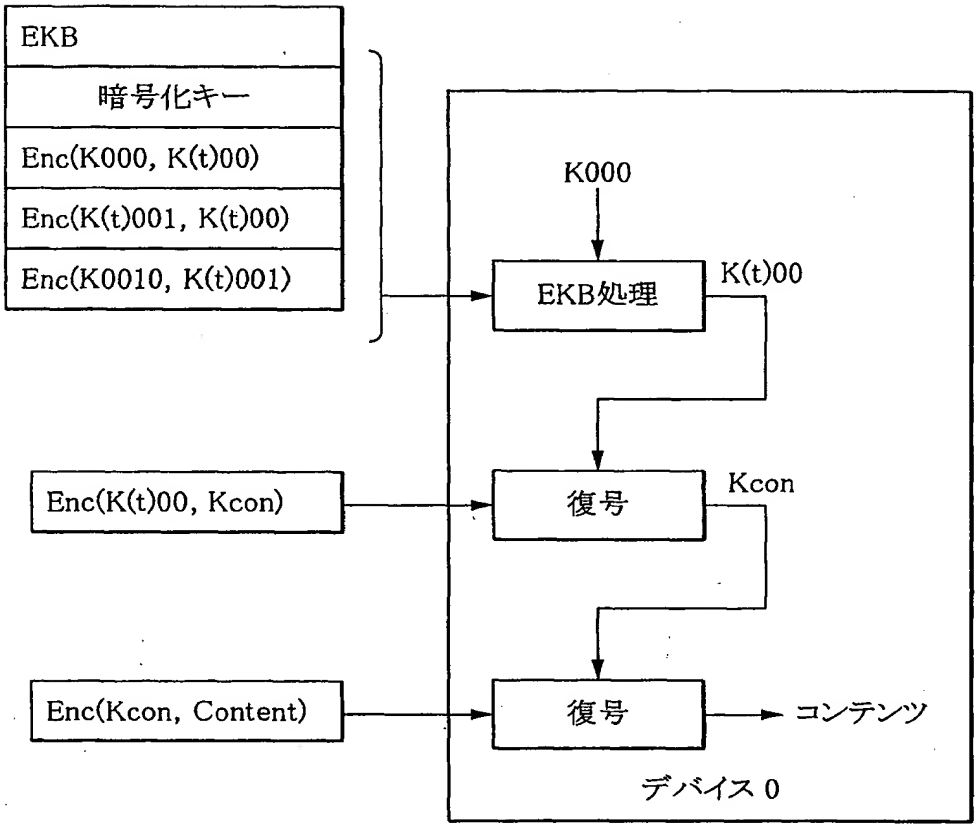

EKB
暗号化キー
Enc(K000, K(t)00)
Enc(K(t)001, K(t)00)
Enc(K0010, K(t)001)
K000
EKB処理
K(t)00
Enc(K(t)00, Kcon)
復号
Kcon
Enc(Kcon, Content)
復号
コンテンツ
デバイス 0

FIG.9

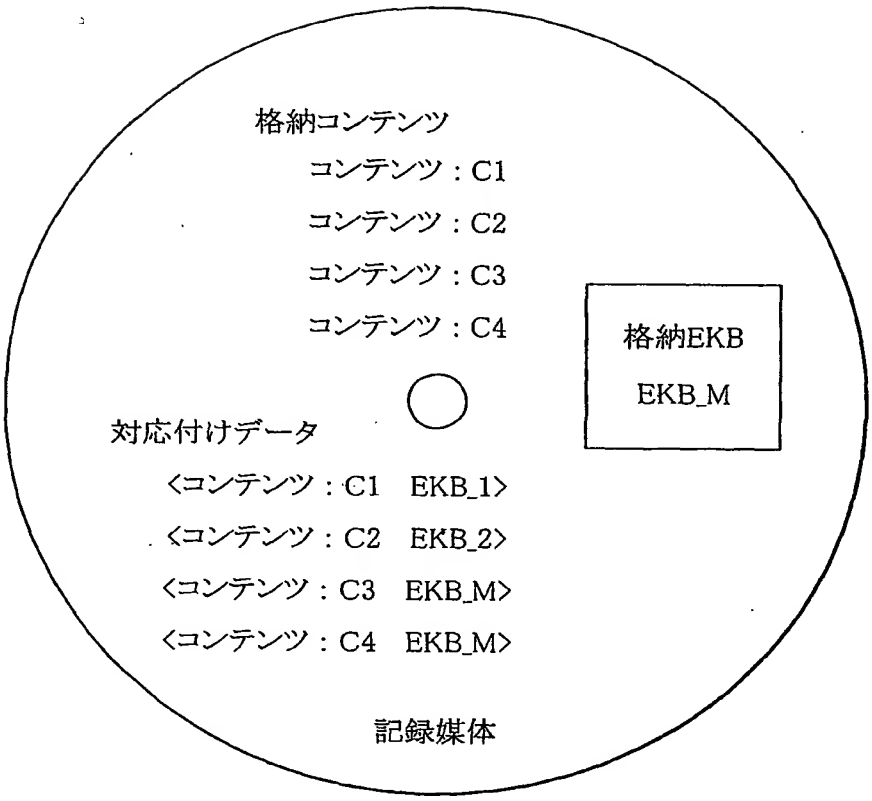

格納コンテンツ
コンテンツ：C1
コンテンツ：C2
コンテンツ：C3
コンテンツ：C4
格納EKB
EKB_M
対応付けデータ
<コンテンツ：C1 EKB_1>
<コンテンツ：C2 EKB_2>
<コンテンツ：C3 EKB_M>
<コンテンツ：C4 EKB_M>
記録媒体

FIG.10

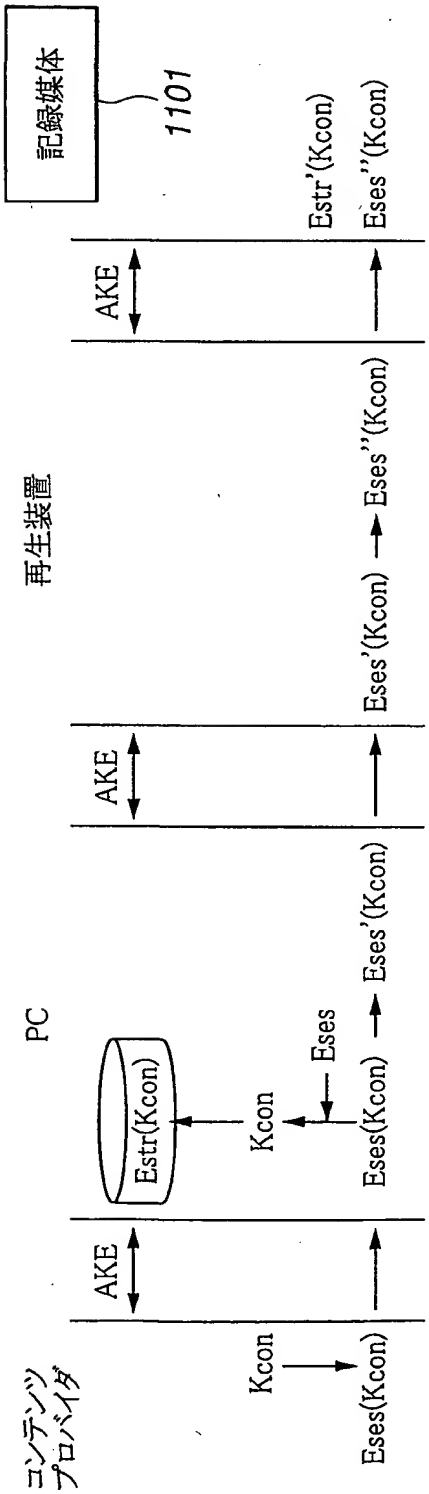


FIG.11A

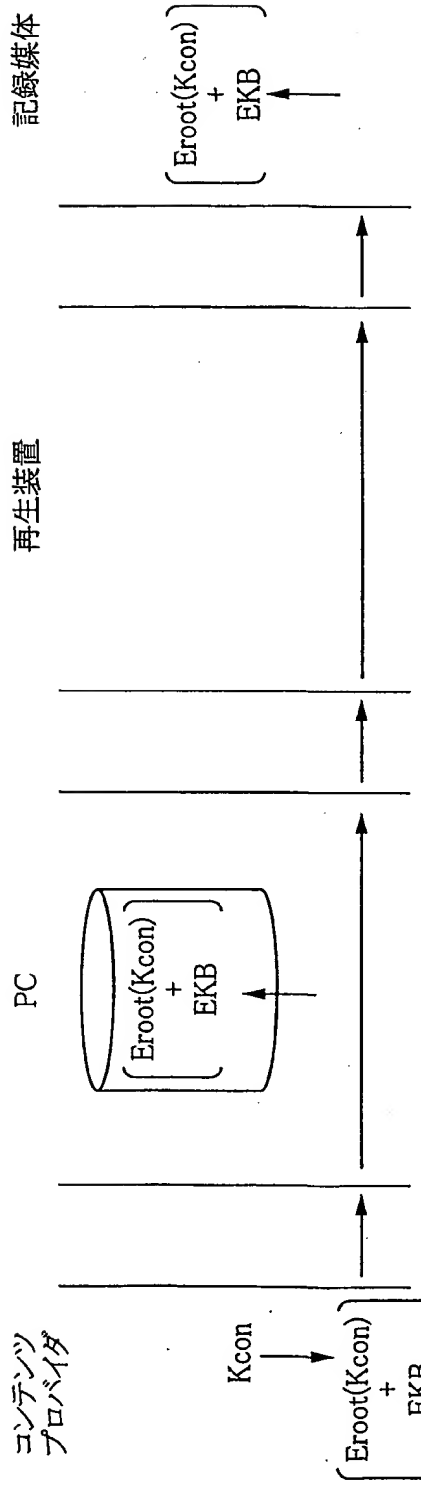


FIG.11B

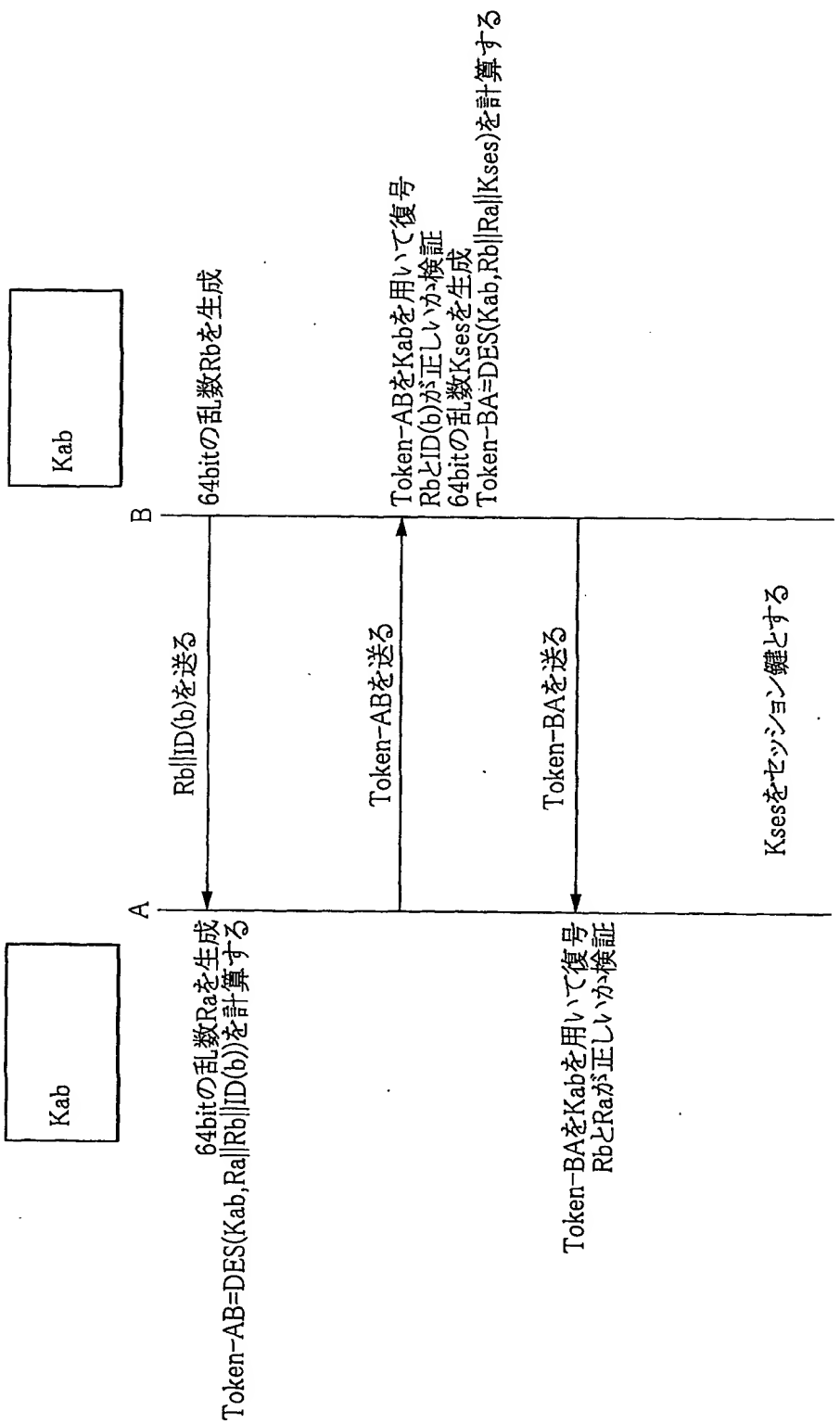

Kab
Kab
A
B
64bitの乱数Raを生成
Token-AB=DES(Kab,Ra||Rb||ID(b))を計算する
Rb||ID(b)を送る
64bitの乱数Rbを生成
Token-ABを送る
Token-ABをKabを用いて復号
RbとID(b)が正しいか検証
64bitの乱数Ksesを生成
Token-BA=DES(Kab,Rb||Ra||Kses)を計算する
Token-BAを送る
Token-BAをKabを用いて復号
RbとRaが正しいか検証
Ksesをセッション鍵とする

FIG.12

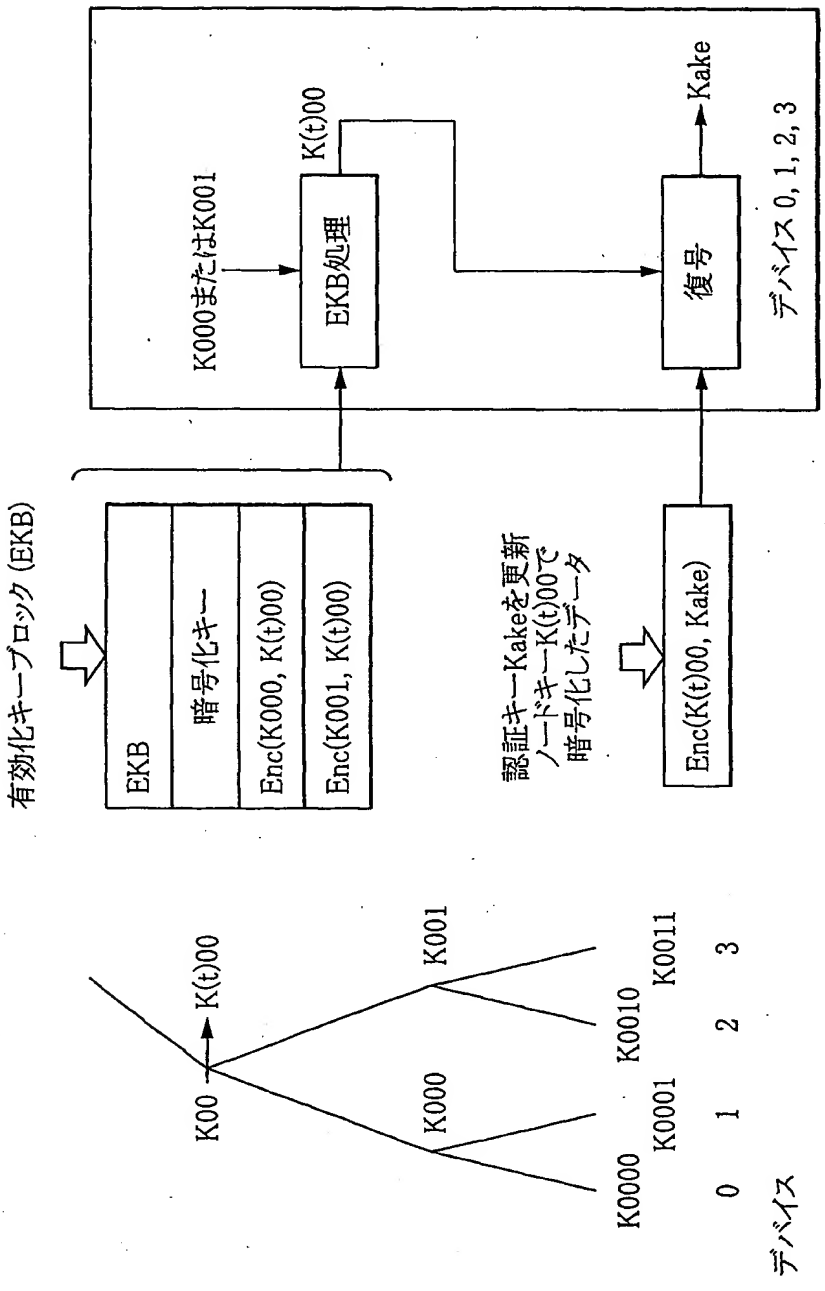

有効化キーブロック (EKB)
EKB
暗号化キー
Enc(K000, K(t)00)
Enc(K001, K(t)00)
K000またはK001
EKB処理
K(t)00
復号
Kake
デバイス 0, 1, 2, 3
認証キーKakeを更新
ノードキーK(t)00で
暗号化したデータ
Enc(K(t)00, Kake)
K00
K(t)00
K000
K001
K0000
K0001
K0010
K0011
0
1
2
3
デバイス

FIG.13

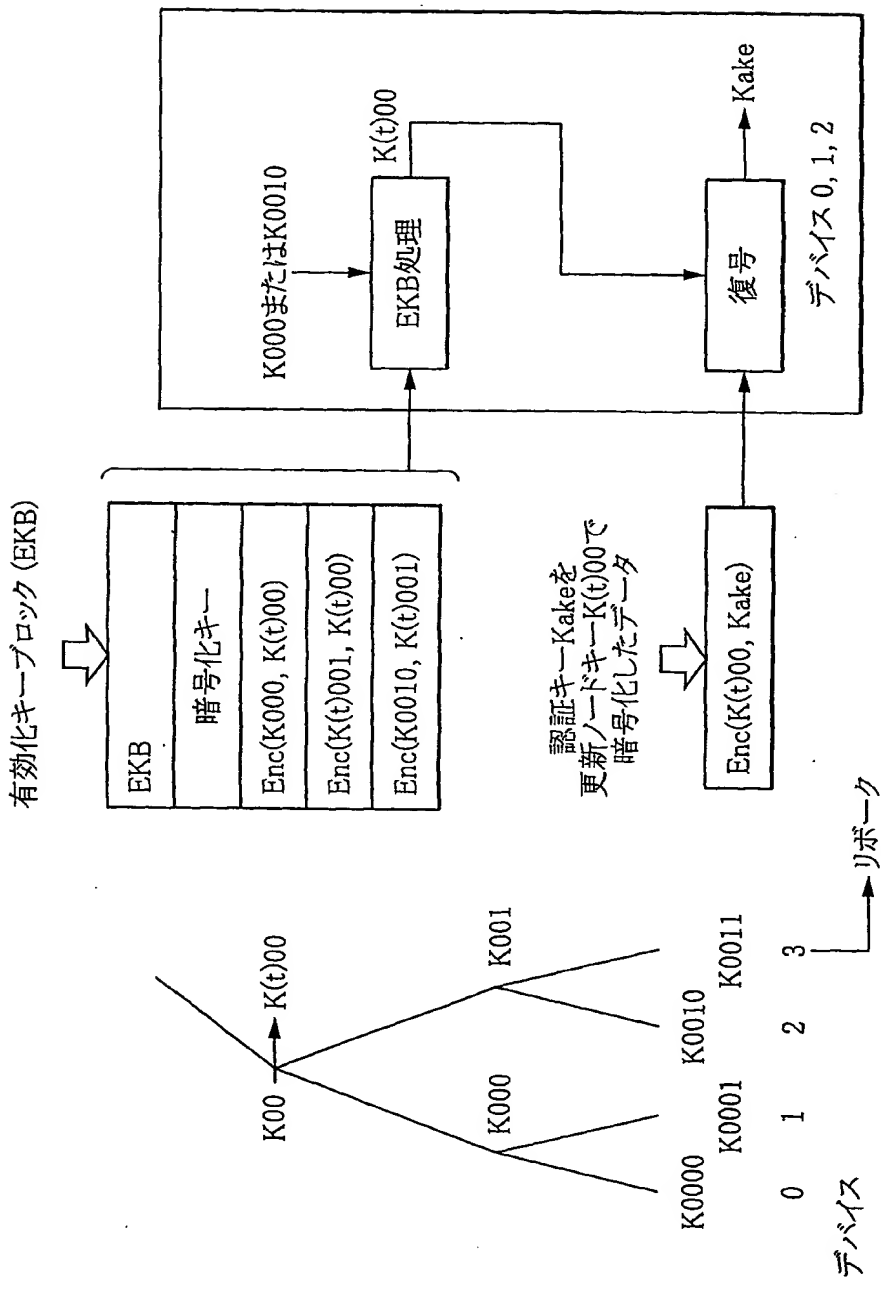

有効化キーブロック (EKB)
EKB
暗号化キー
Enc(K000, K(t)00)
Enc(K(t)001, K(t)00)
Enc(K0010, K(t)001)
K000またはK0010
EKB処理
K(t)00
Kake
復号
デバイス 0, 1, 2
認証キーKakeを
更新ノードキーK(t)00で
暗号化したデータ
Enc(K(t)00, Kake)
K00
K(t)00
K000
K001
K0000
K0001
K0010
K0011
0
1
2
3
リボーク
デバイス

FIG.14

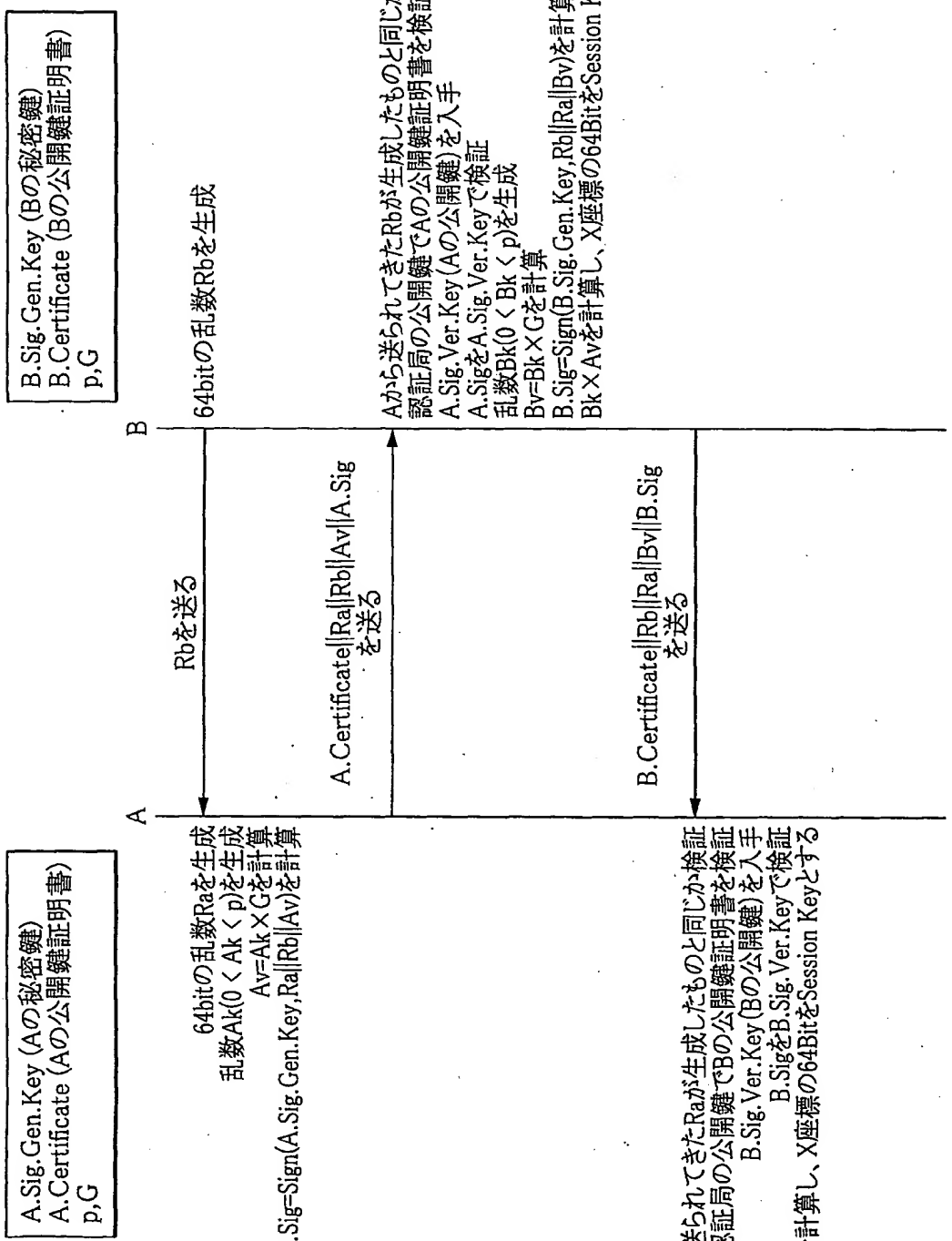

A.Sig.Gen.Key (Aの秘密鍵)
A.Certificate (Aの公開鍵証明書)
p,G
B.Sig.Gen.Key (Bの秘密鍵)
B.Certificate (Bの公開鍵証明書)
p,G
A
B
64bitの乱数Raを生成
乱数Ak(0 < Ak < p)を生成
Av=Ak×Gを計算
A.Sig=Sign(A.Sig.Gen.Key,Ra||Rb||Av)を計算
Rbを送る
64bitの乱数Rbを生成
A.Certificate||Ra||Rb||Av||A.Sigを送る
Aから送られてきたRbが生成したものと同じか検証
認証局の公開鍵でAの公開鍵証明書を検証
A.Sig.Ver.Key(Aの公開鍵)を入手
A.SigをA.Sig.Ver.Keyで検証
乱数Bk(0 < Bk < p)を生成
Bv=Bk×Gを計算
B.Sig=Sign(B.Sig.Gen.Key,Rb||Ra||Bv)を計算
Bk×Avを計算し、X座標の64BitをSession Keyとする
B.Certificate||Rb||Ra||Bv||B.Sigを送る
Bから送られてきたRaが生成したものと同じか検証
認証局の公開鍵でBの公開鍵証明書を検証
B.Sig.Ver.Key(Bの公開鍵)を入手
B.SigをB.Sig.Ver.Keyで検証
Ak×Bvを計算し、X座標の64BitをSession Keyとする

FIG.15

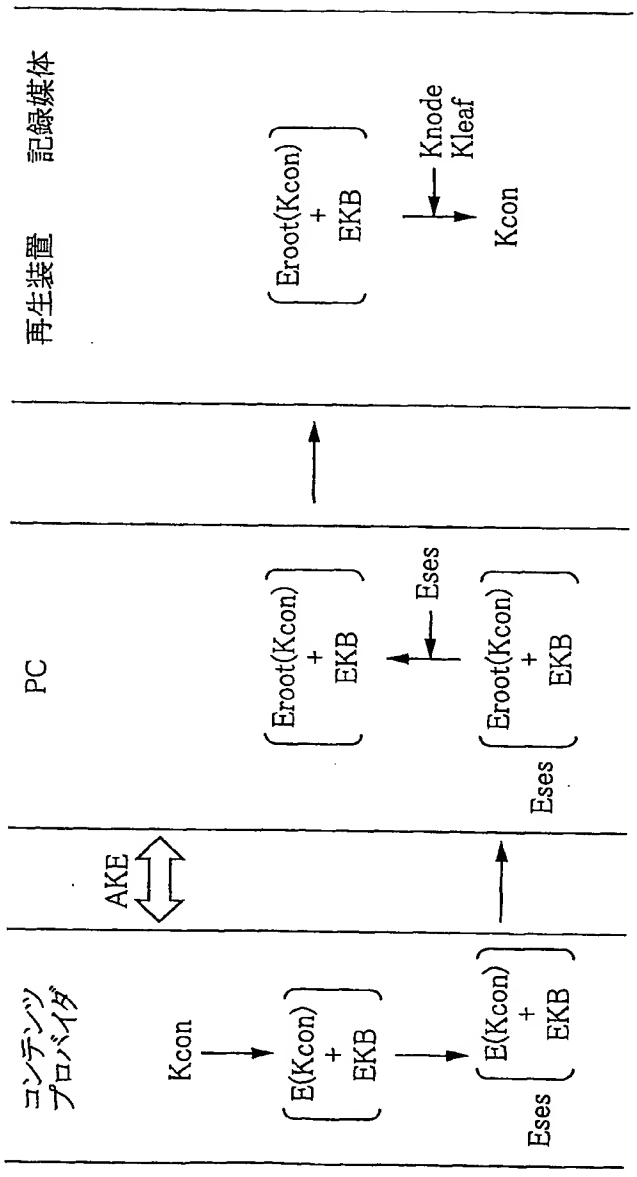

コンテンツプロバイダ
PC
再生装置　記録媒体
Kcon
E(Kcon) + EKB
Eses
AKE
Eroot(Kcon) + EKB
Ekb
Knode Kleaf
Kcon

FIG.16

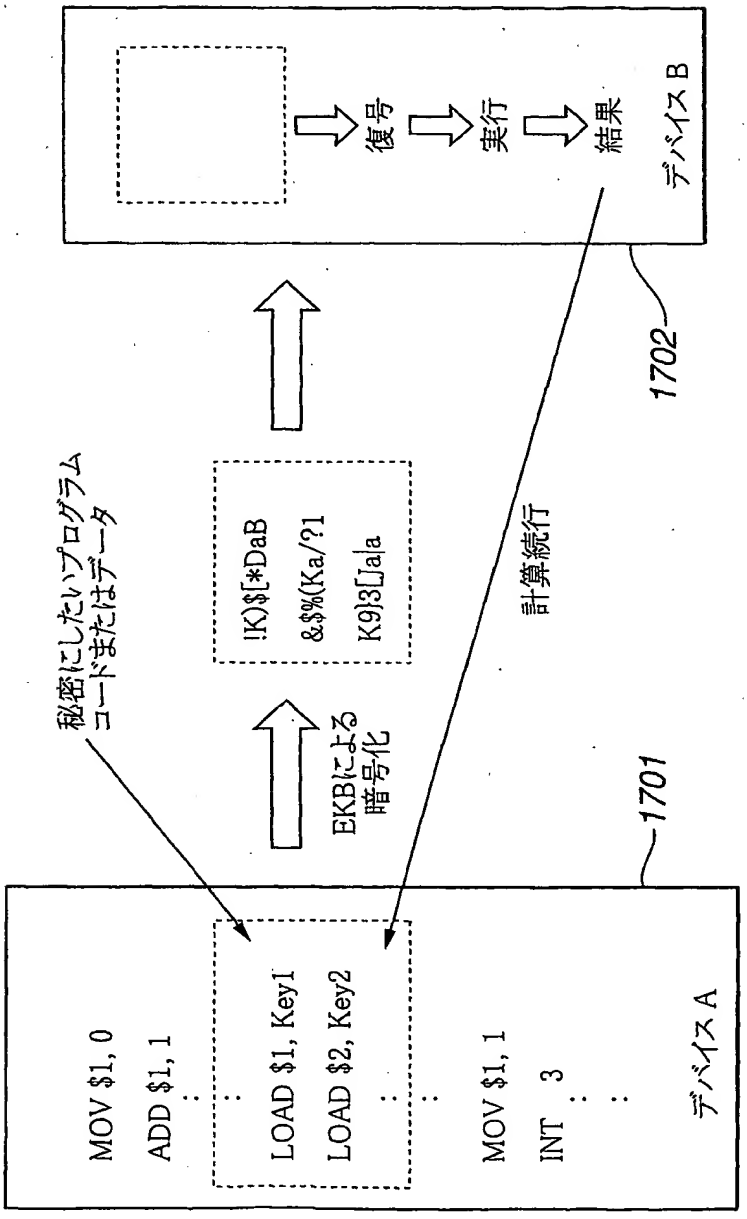

デバイス A
MOV $1, 0
ADD $1, 1
LOAD $1, Key1
LOAD $2, Key2
MOV $1, 1
INT 3
1701
EKBによる暗号化
!K)$[*DaB
&$%(Ka/?1
K9)3[Ja|a
秘密にしたいプログラムコードまたはデータ
計算続行
デバイス B
復号
実行
結果
1702

FIG.17

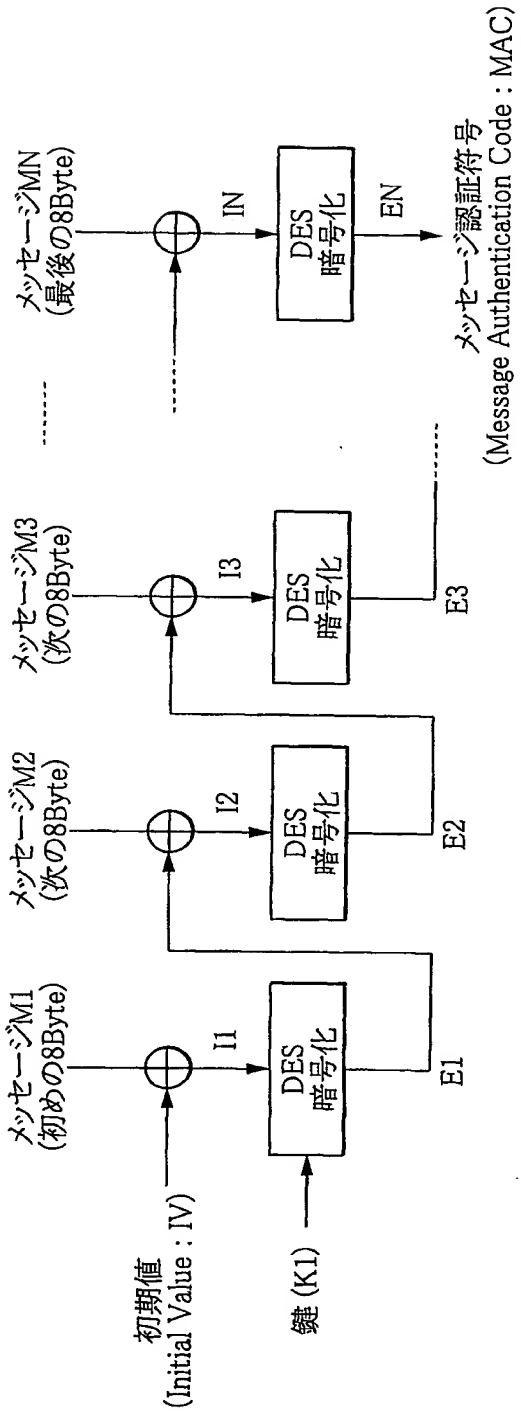

メッセージM1
(初めの8Byte)
メッセージM2
(次の8Byte)
メッセージM3
(次の8Byte)
メッセージMN
(最後の8Byte)
初期値
(Initial Value : IV)
鍵 (K1)
DES
暗号化
I1
I2
I3
IN
E1
E2
E3
EN
メッセージ認証符号
(Message Authentication Code : MAC)
⊕ : 非他的論理和処理 (8バイト単位)

FIG.18

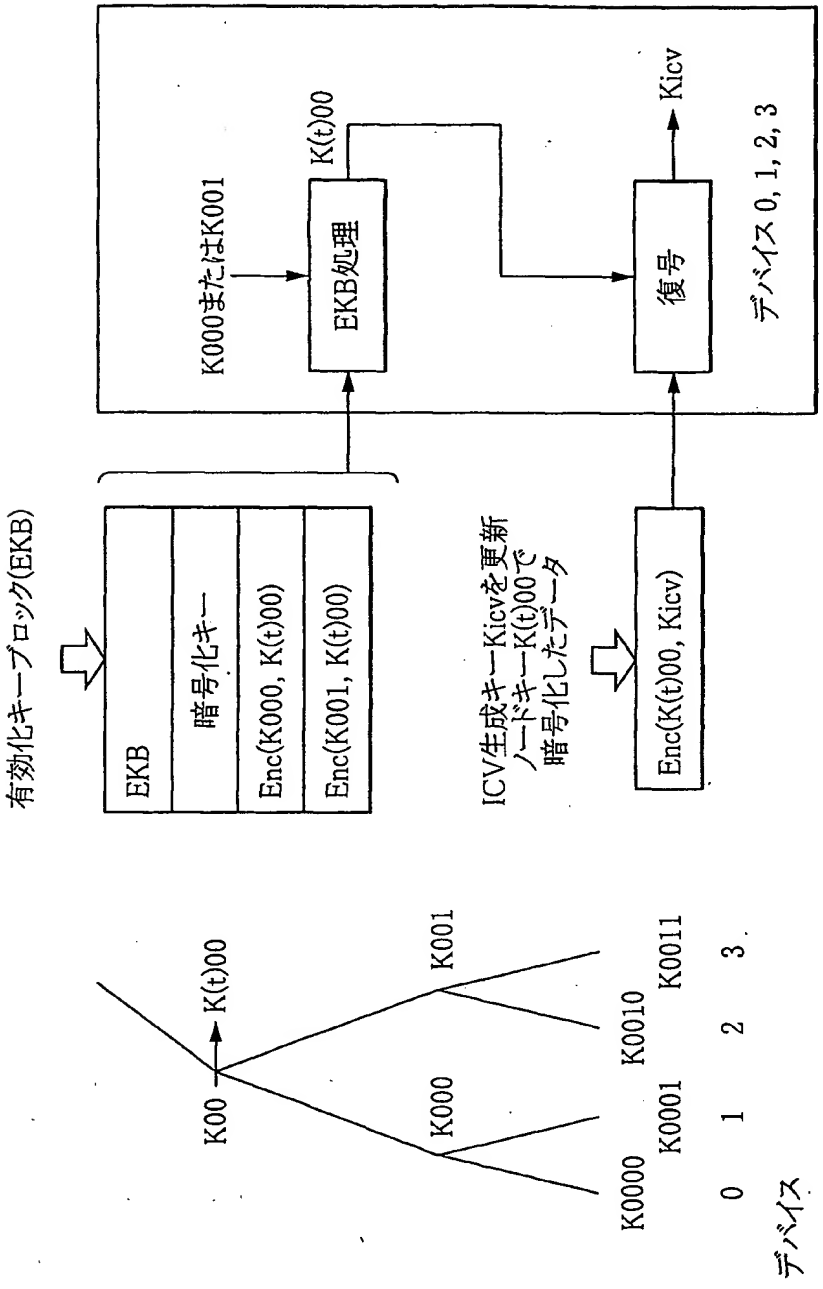

有効化キーブロック(EKB)
EKB
暗号化キー
Enc(K000, K(t)00)
Enc(K001, K(t)00)
K000またはK001
EKB処理
K(t)00
ICV生成キーKicvを更新
ノードキーK(t)00で
暗号化したデータ
Enc(K(t)00, Kicv)
復号
Kicv
デバイス 0, 1, 2, 3
K00
K(t)00
K000
K001
K0000
K0001
K0010
K0011
0
1
2
3
デバイス

FIG.19

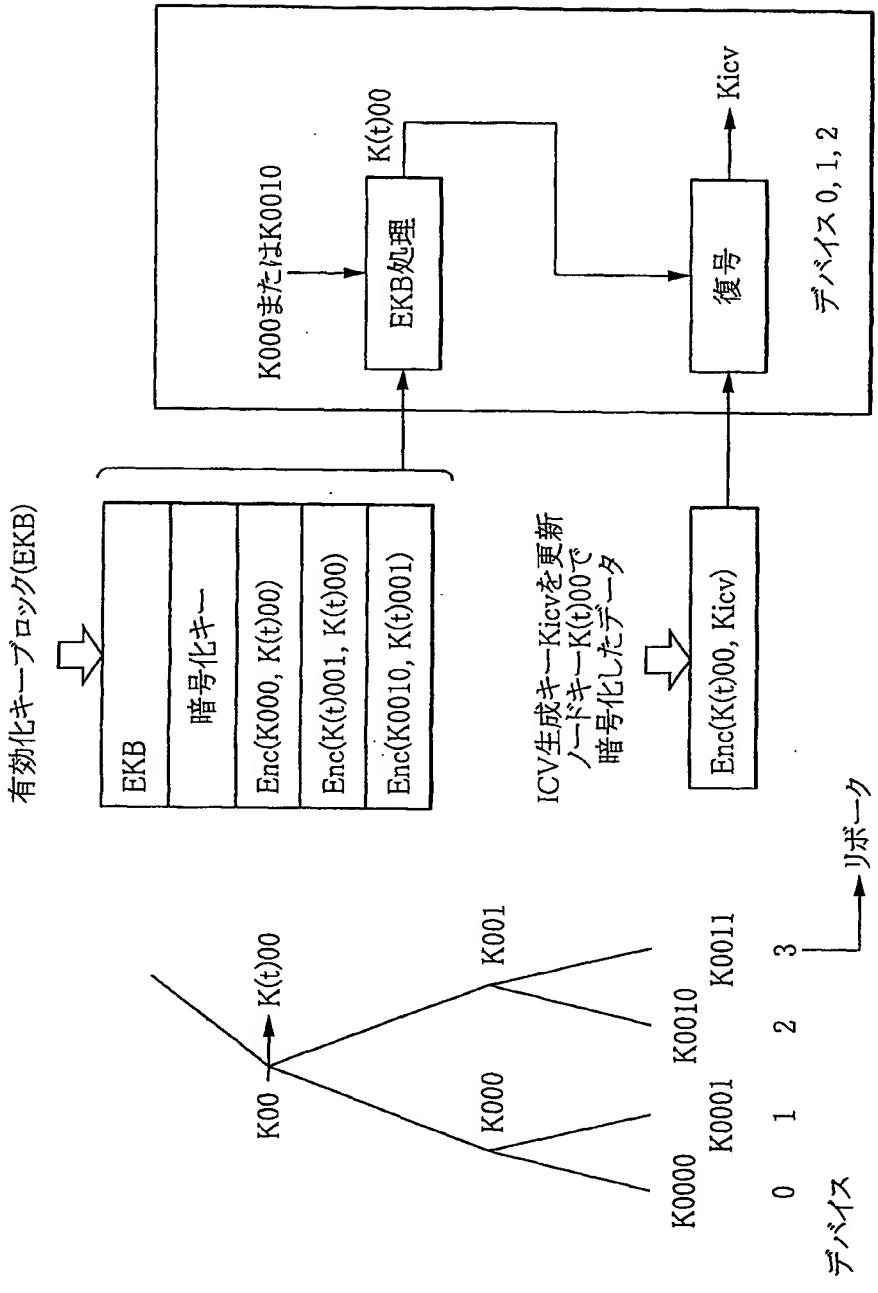

有効化キーブロック(EKB)
EKB
暗号化キー
Enc(K000, K(t)00)
Enc(K(t)001, K(t)00)
Enc(K0010, K(t)001)
K000またはK0010
EKB処理
K(t)00
復号
Kicv
デバイス 0, 1, 2
ICV生成キーKicvを更新ノードキーK(t)00で暗号化したデータ
Enc(K(t)00, Kicv)
K00
K(t)00
K000
K001
K0000
K0001
K0010
K0011
0
1
2
3
リボーク
デバイス

FIG.20

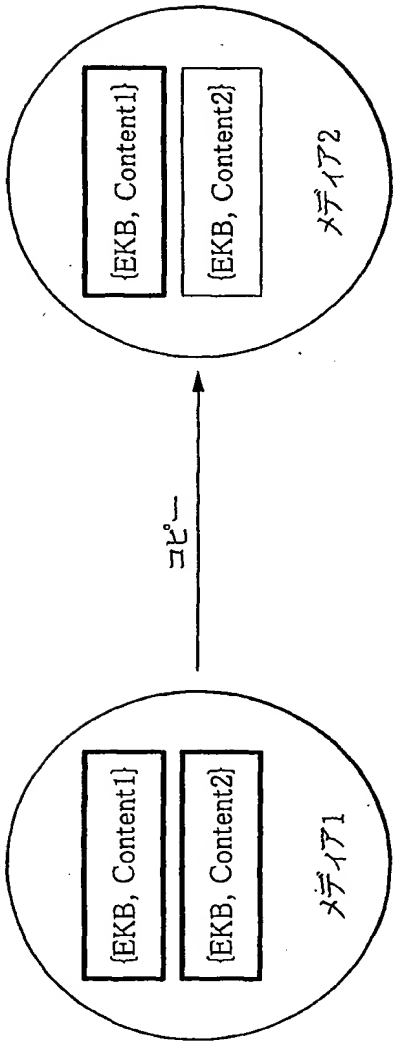


FIG.21A

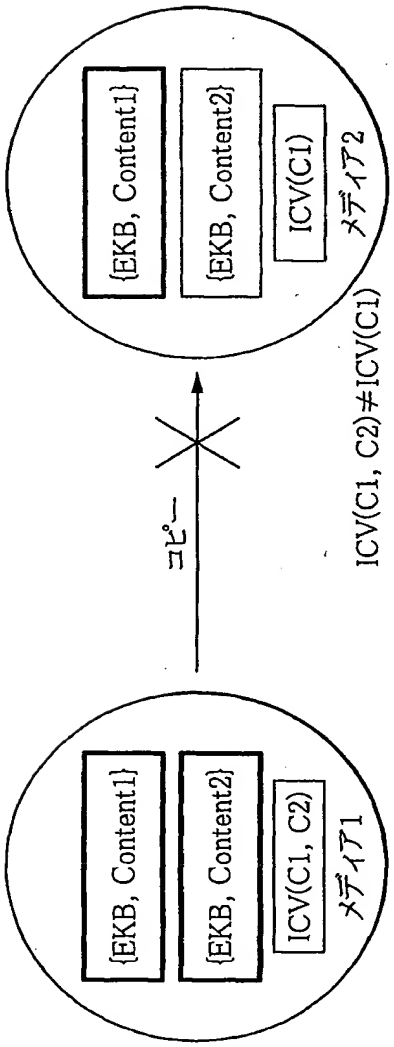


FIG.21B

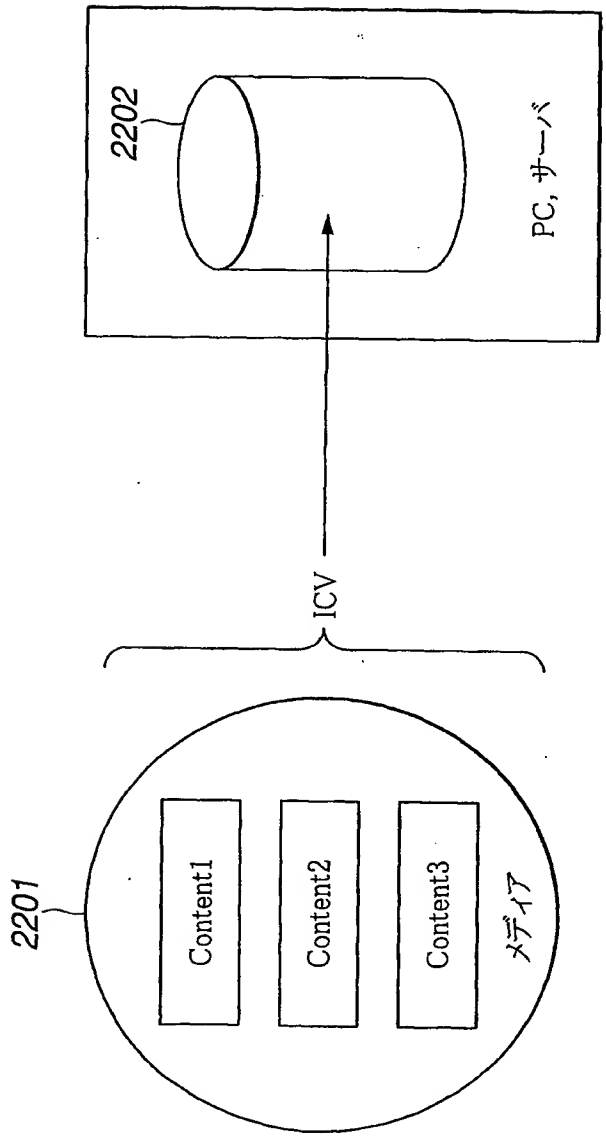


FIG.22

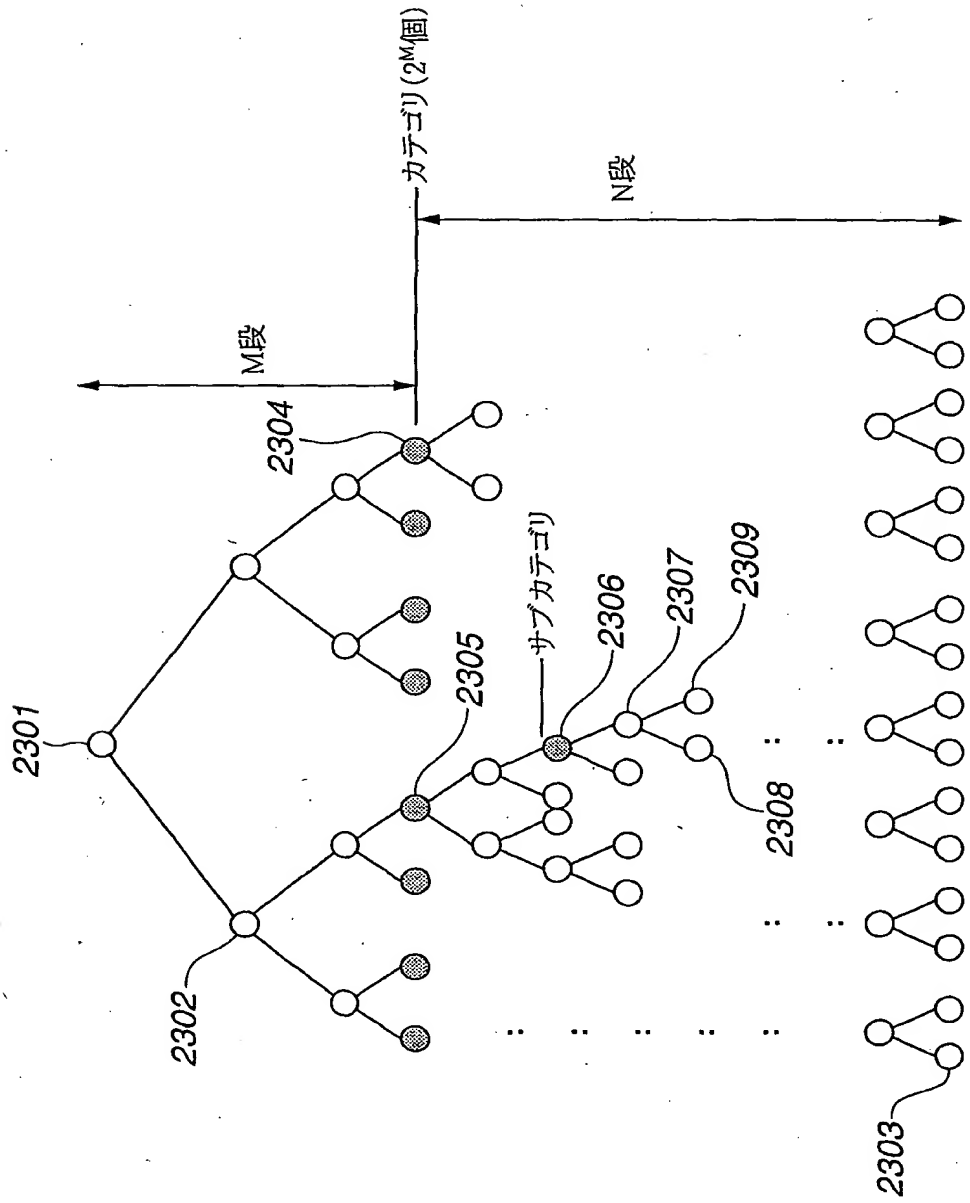

M段
カテゴリ (2^M個)
N段
2301
2302
2303
2304
2305
サブカテゴリ
2306
2307
2308
2309

FIG.23

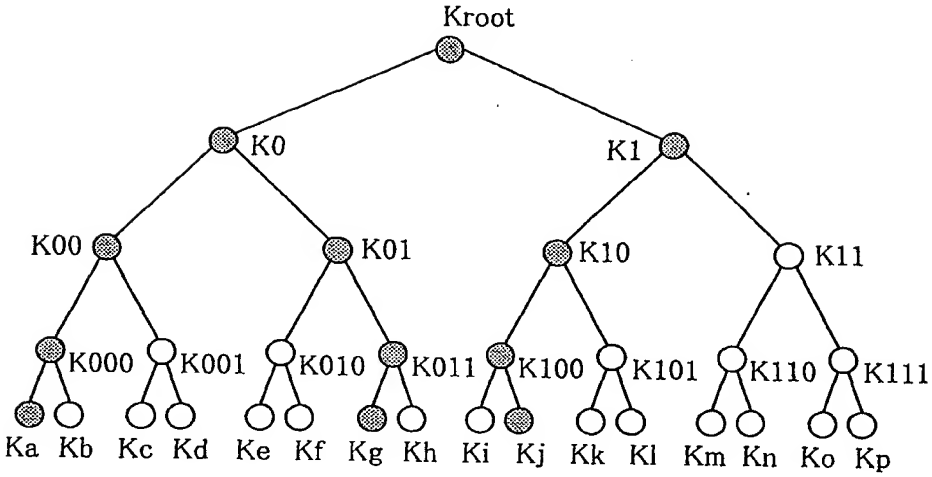


FIG.24A

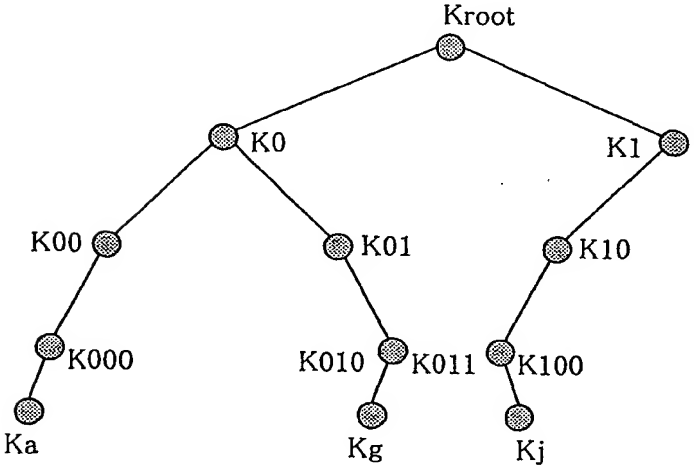


FIG.24B

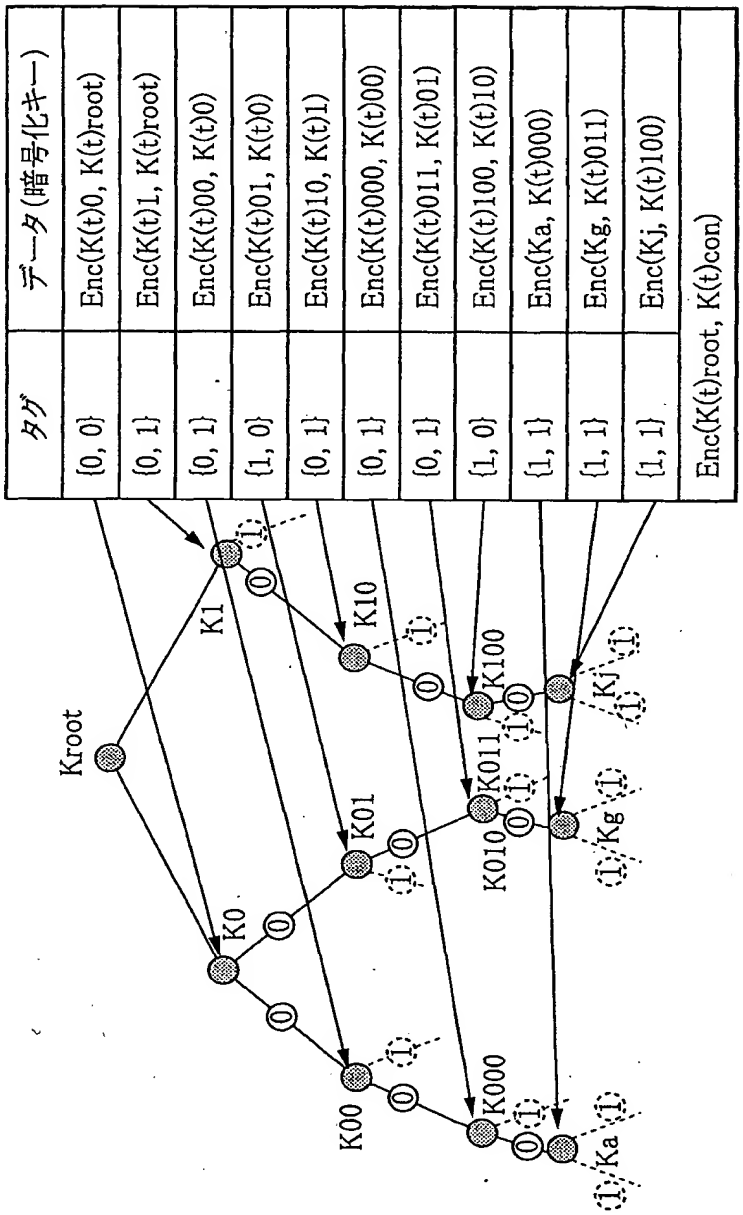


FIG.25A

FIG.25B

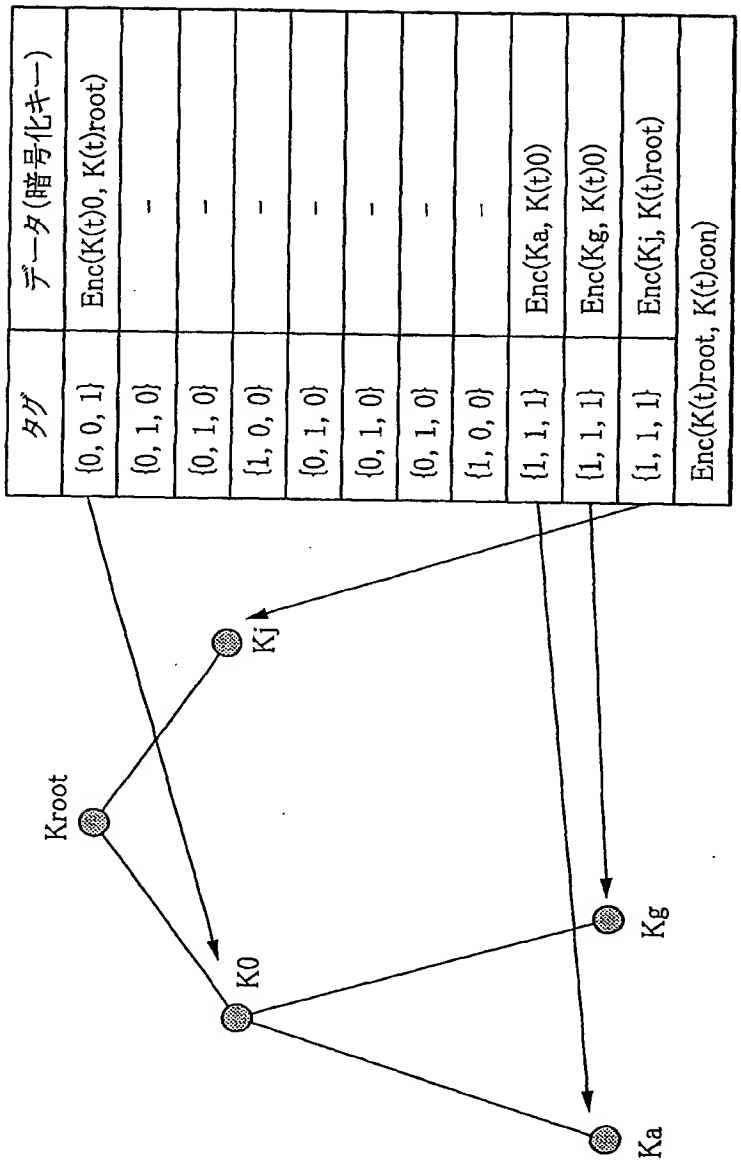


FIG.26A

FIG.26B

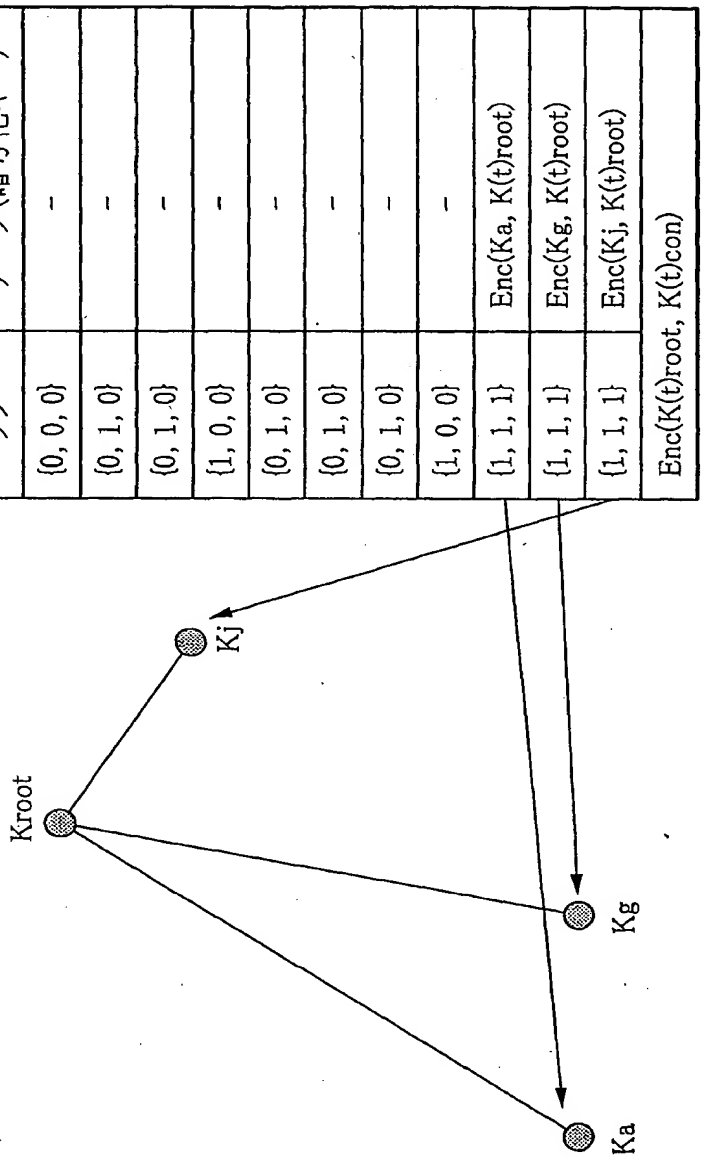


FIG.27A

FIG.27B

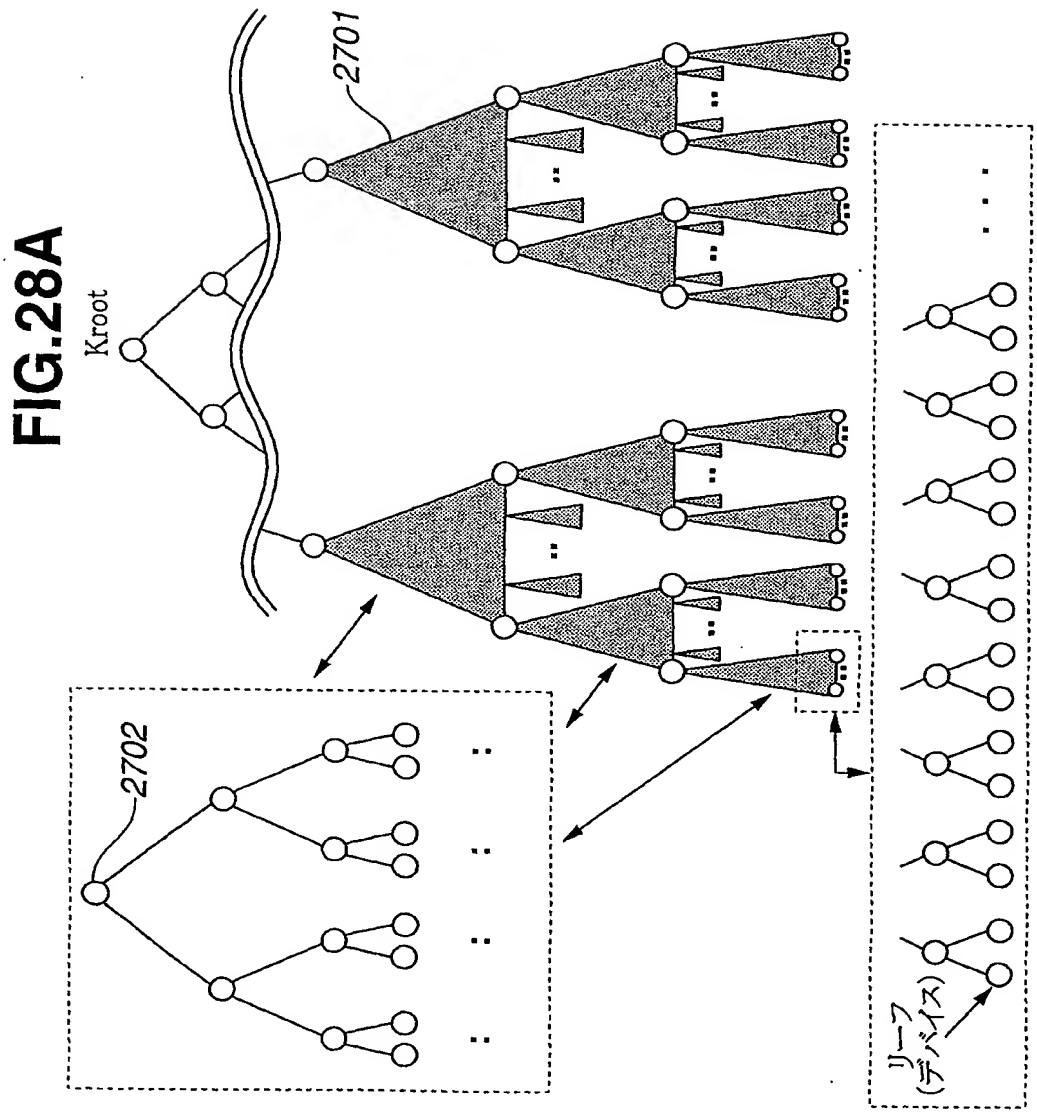


FIG.28A

FIG.28B

FIG.28C

FIG.29A

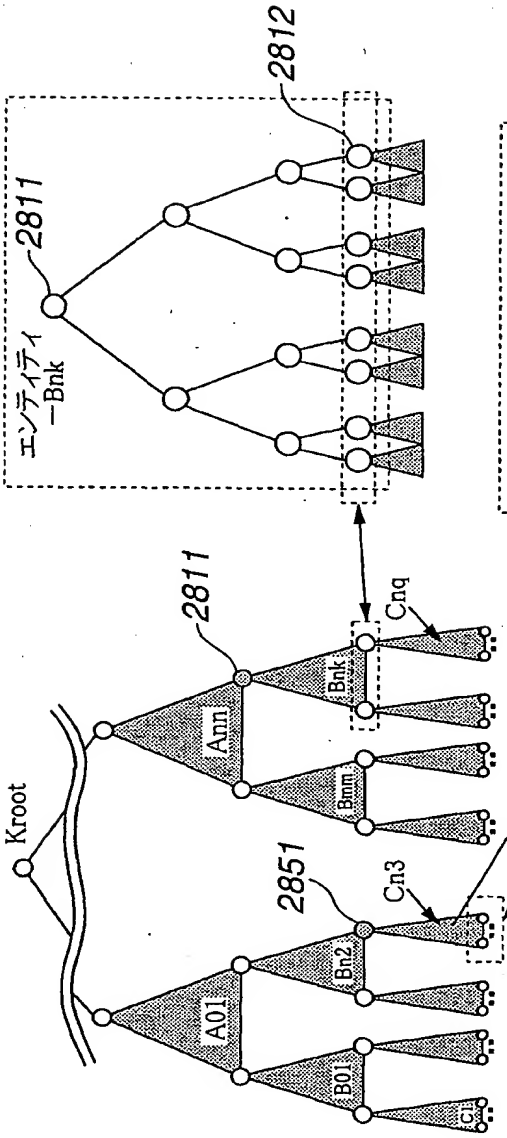
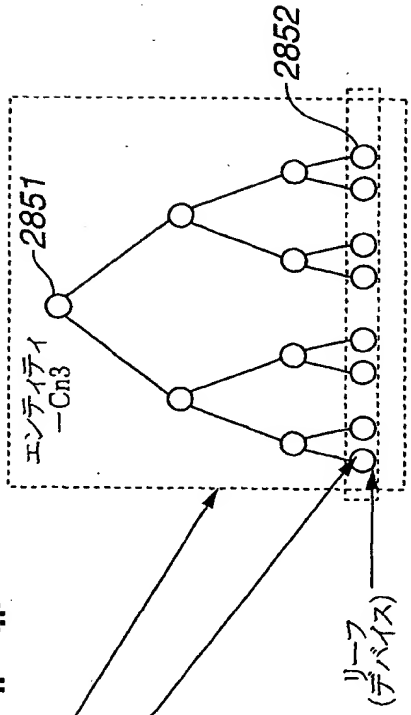


FIG.29B

FIG.29C



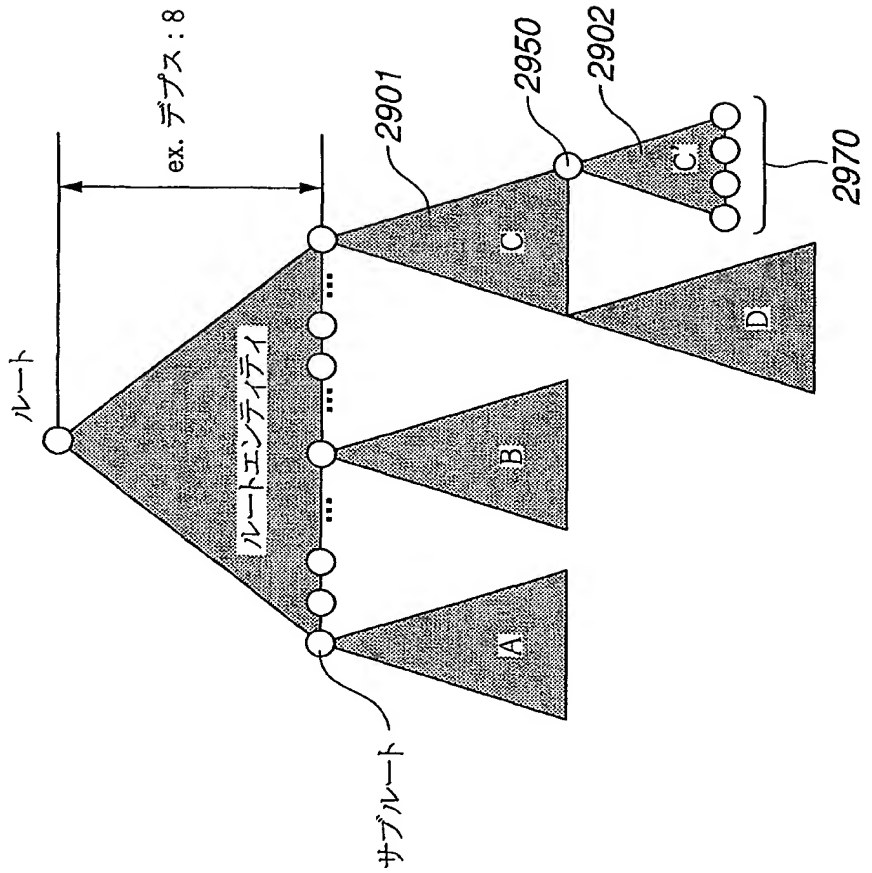


FIG.30B

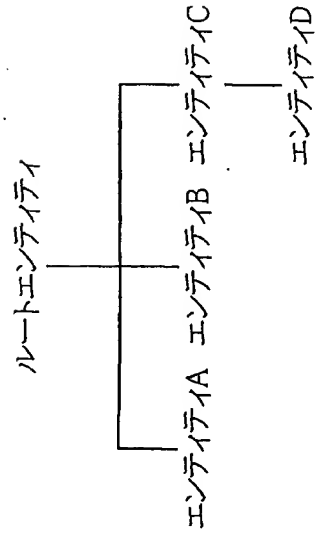


FIG.30A

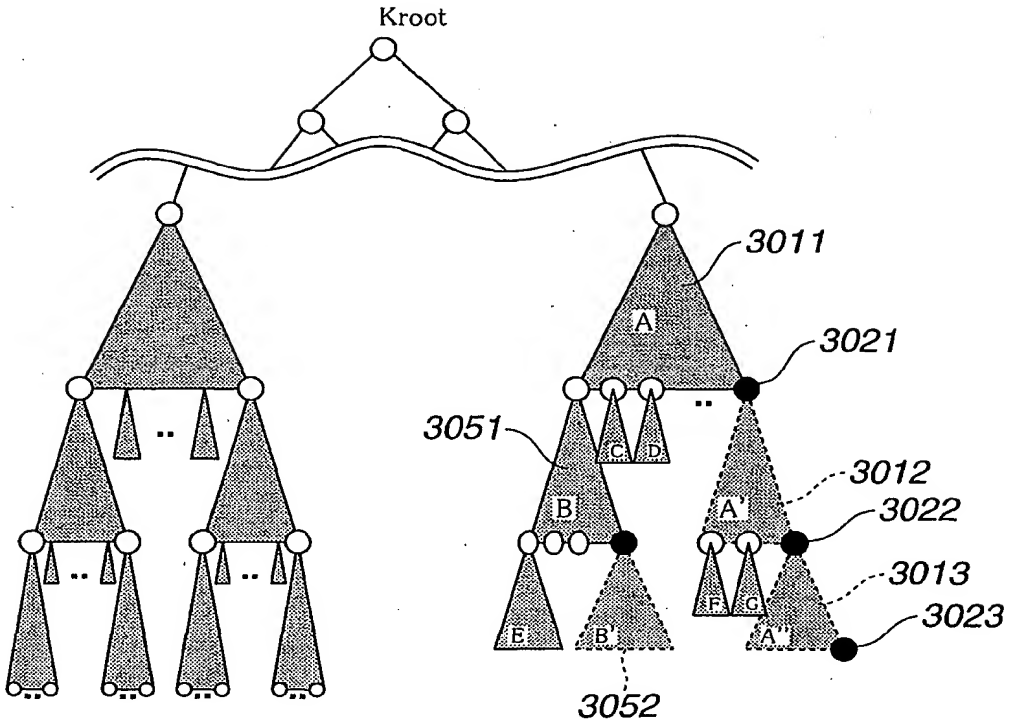


FIG.31

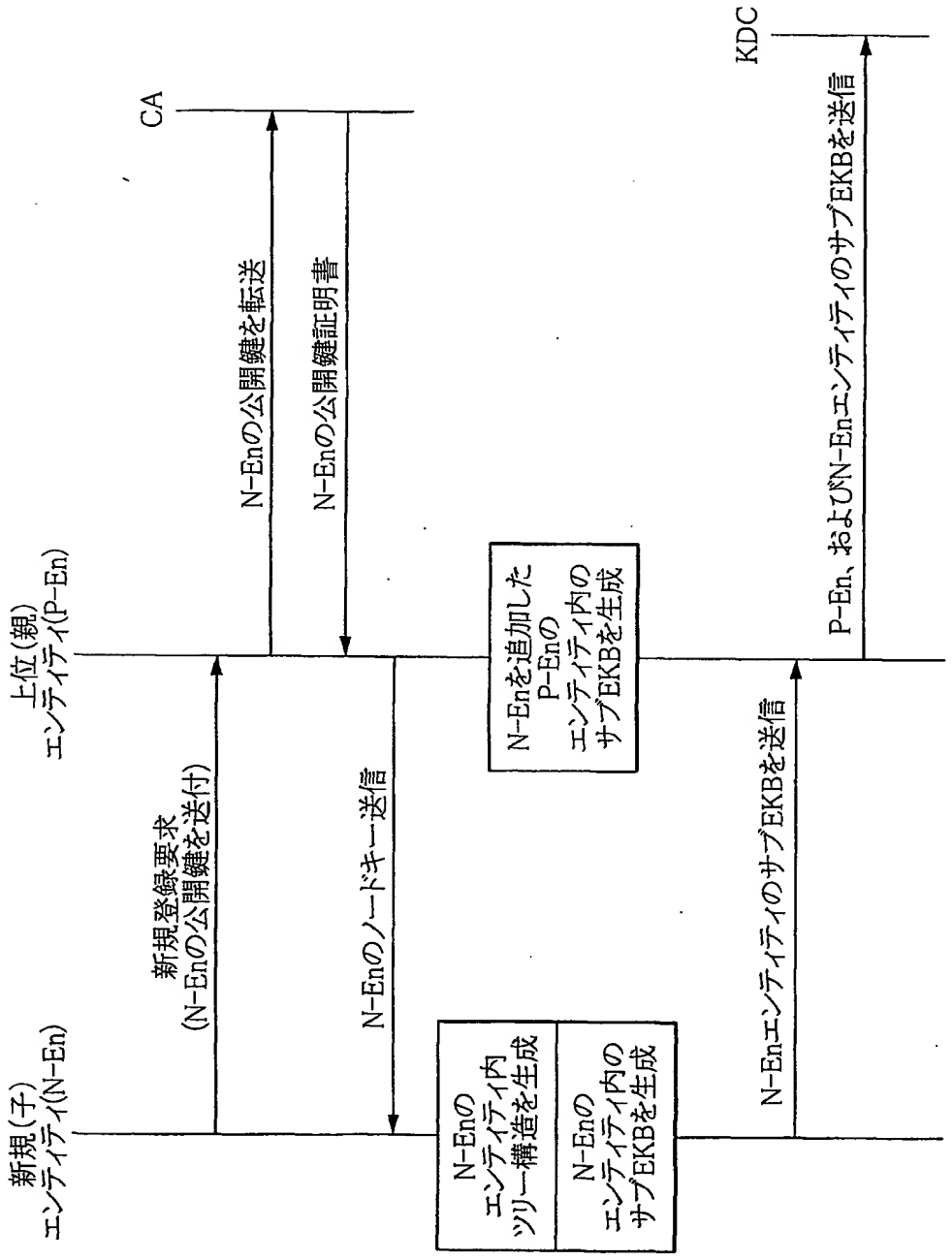

新規(子)
エンティティ(N-En)
上位(親)
エンティティ(P-En)
CA
KDC
新規登録要求
(N-Enの公開鍵を送付)
N-Enの公開鍵を転送
N-Enの公開鍵証明書
N-Enのノードキー送信
N-Enを追加した
P-Enの
エンティティ内の
サブEKBを生成
P-En、およびN-Enエンティティのサブ EKBを送信
N-Enの
エンティティ内
ツリー構造を生成
N-Enの
エンティティ内の
サブEKBを生成
N-Enエンティティのサブ EKBを送信

FIG.32

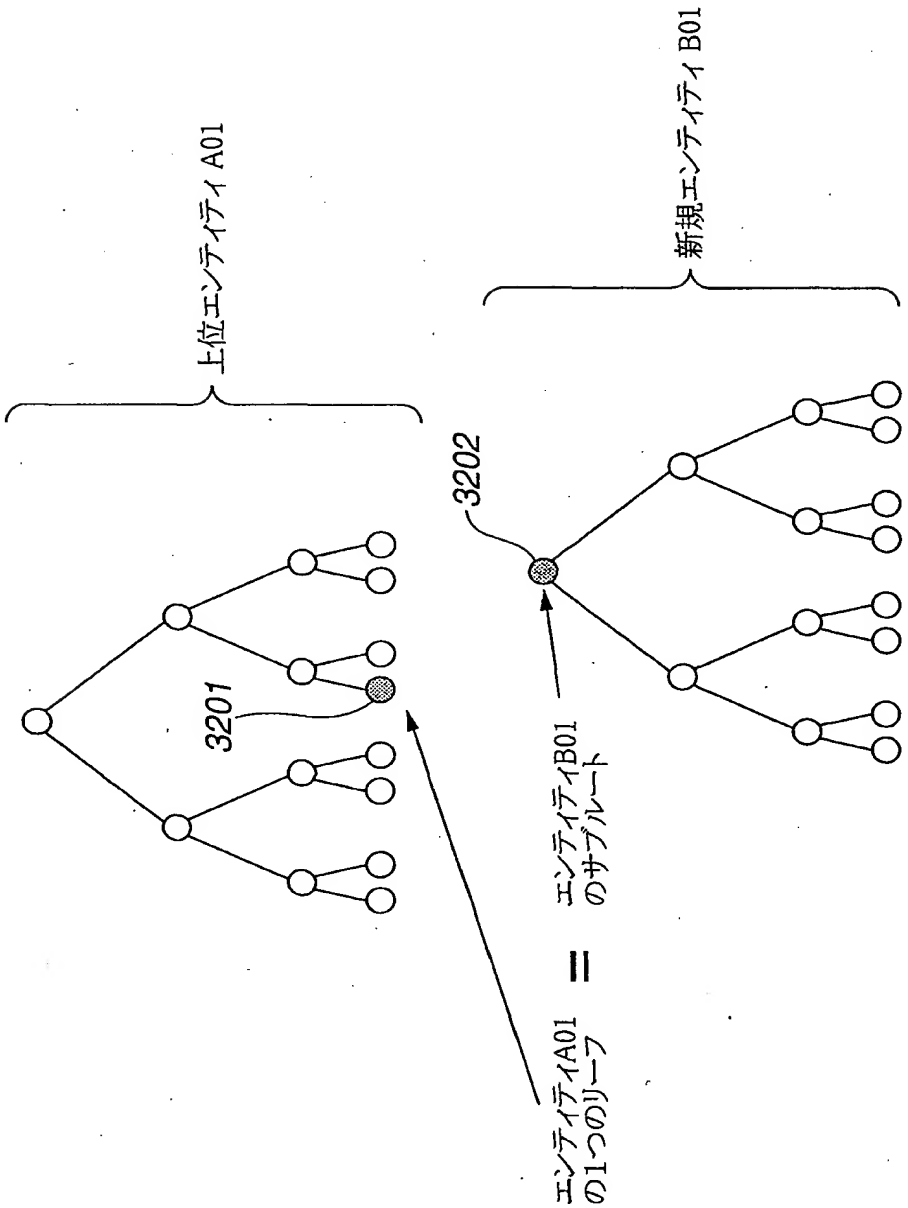

上位エンティティ A01
新規エンティティ B01
3201
3202
エンティティA01
の1つのリーフ
=
エンティティB01
のサブルート

FIG.33

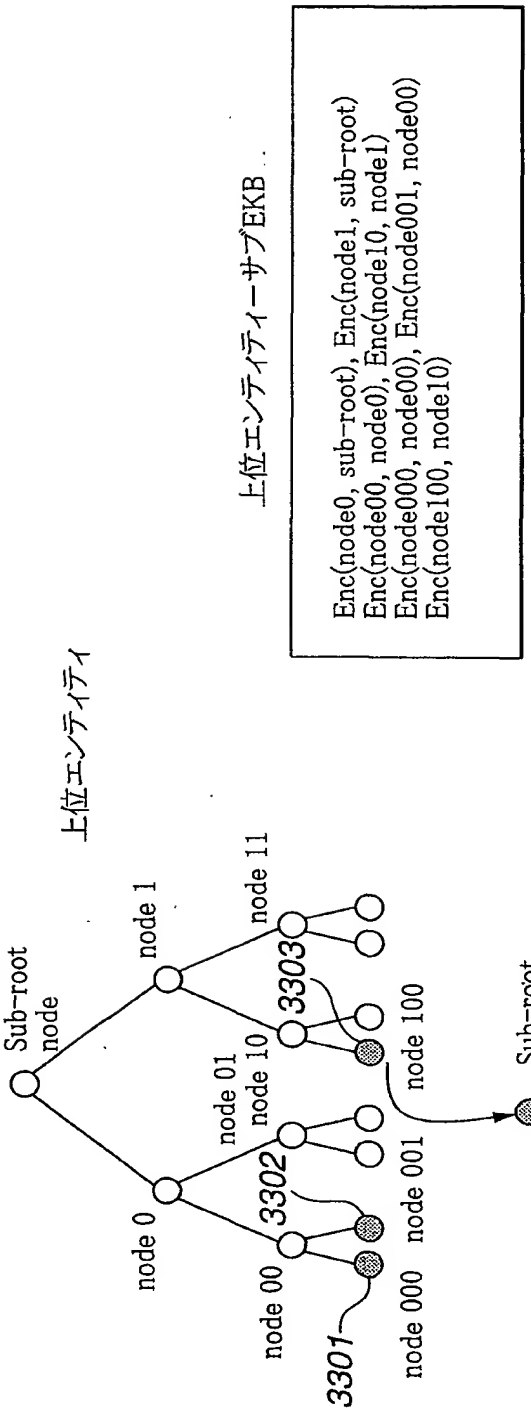


FIG.34A

上位エンティティーサブEKB

Enc(node0, sub-root), Enc(node1, sub-root)
Enc(node00, node0), Enc(node10, node1)
Enc(node000, node00), Enc(node001, node00)
Enc(node100, node10)

FIG.34B

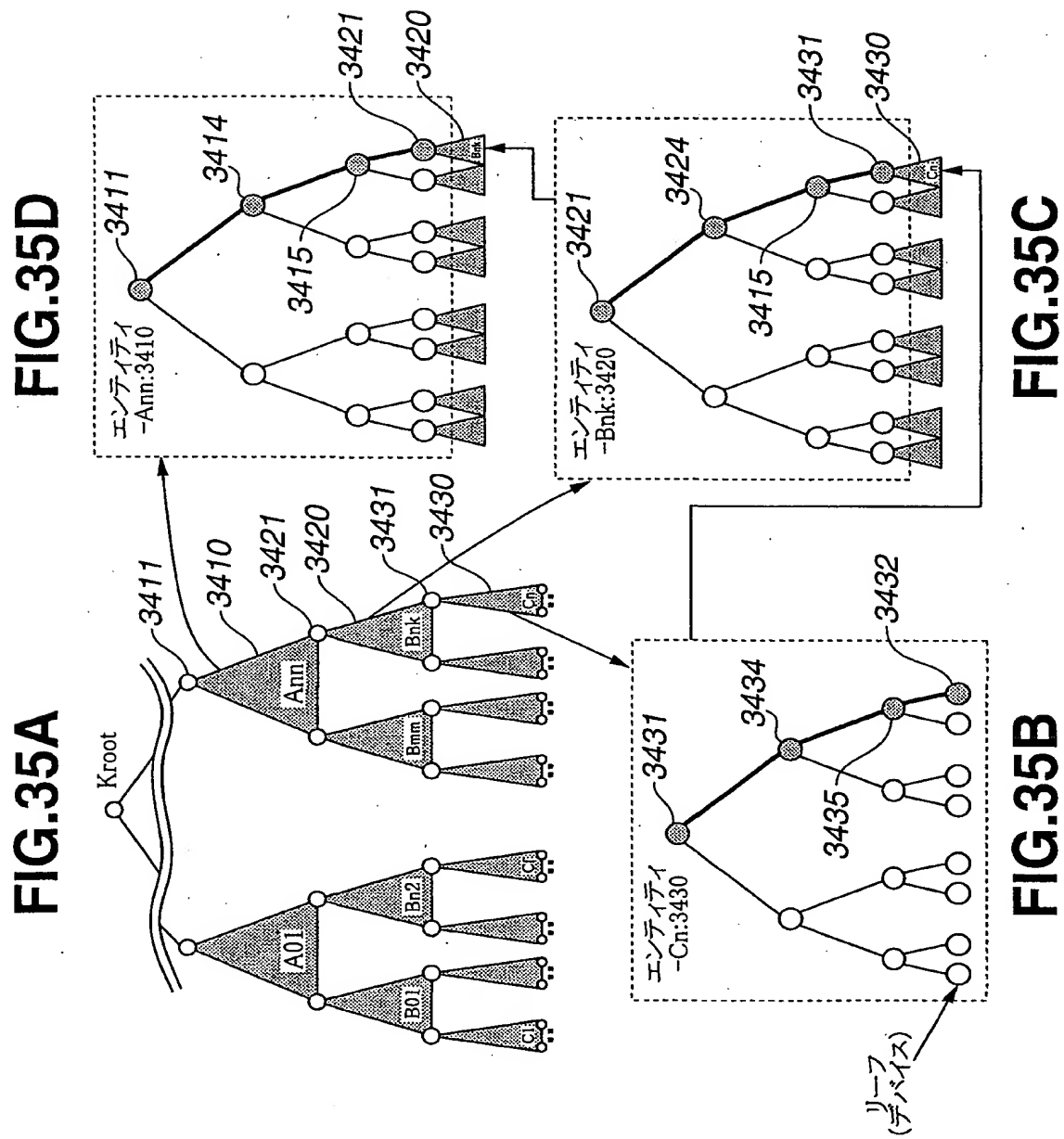


FIG.35A
Kroot
3411
3410
3421
3420
3431
3430
Ann
Bnk
Bmm
Cn
A01
Bn2
C0
B01
C1
FIG.35D
エンティティ
-Ann:3410
3411
3414
3415
3421
3420
FIG.35C
エンティティ
-Bnk:3420
3421
3424
3415
3431
3430
FIG.35B
エンティティ
-Cn:3430
3431
3434
3435
3432
リーフ
(デバイス)

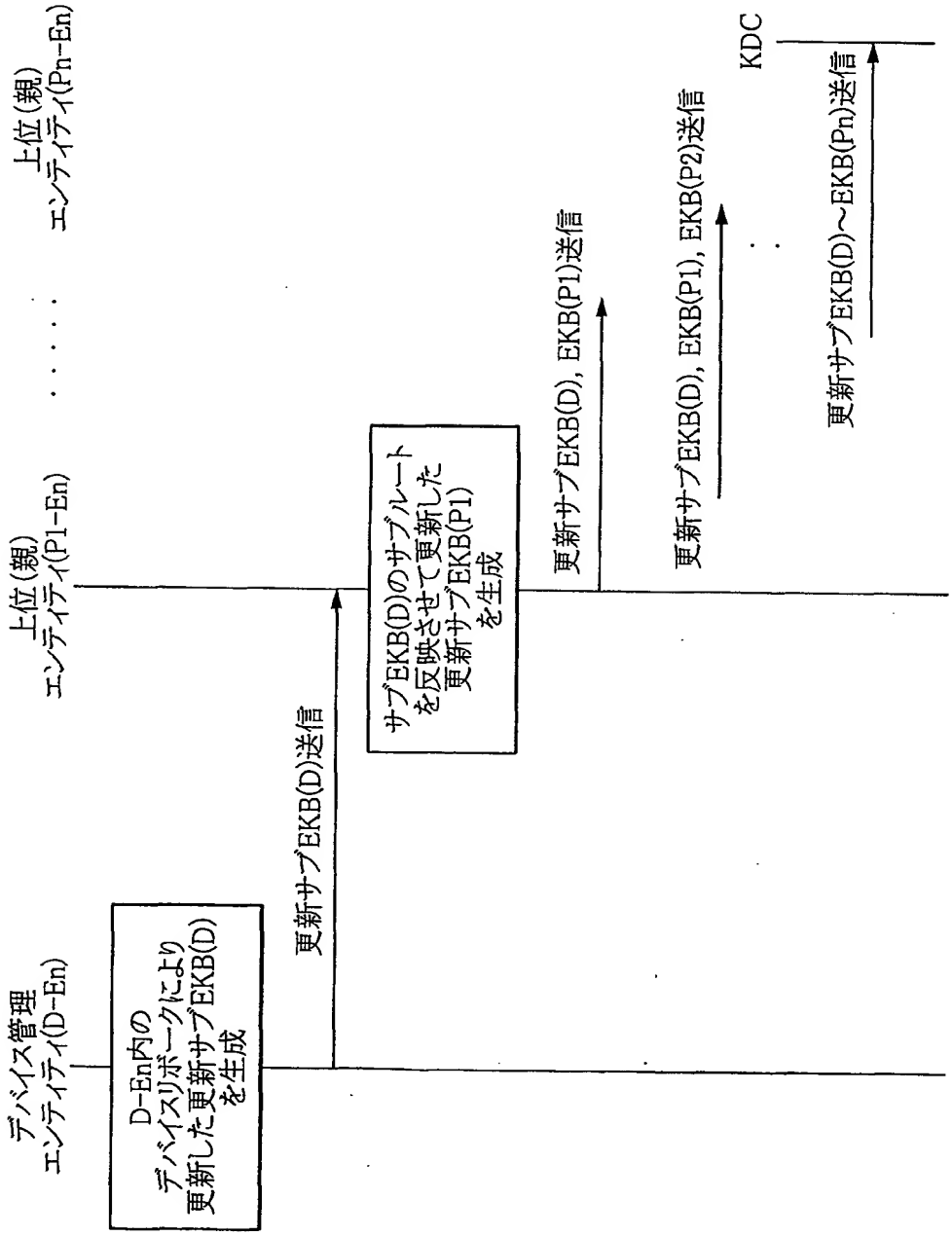

デバイス管理
エンティティ(D-En)
D-En内の
デバイスリポジトリにより
更新した更新サブEKB(D)
を生成
更新サブEKB(D)送信
上位(親)
エンティティ(P1-En)
サブEKB(D)のサブルート
を反映させて更新した
更新サブEKB(P1)
を生成
更新サブEKB(D), EKB(P1)送信
更新サブEKB(D), EKB(P1), EKB(P2)送信
KDC
更新サブEKB(D)～EKB(Pn)送信
上位(親)
エンティティ(Pn-En)

FIG.36

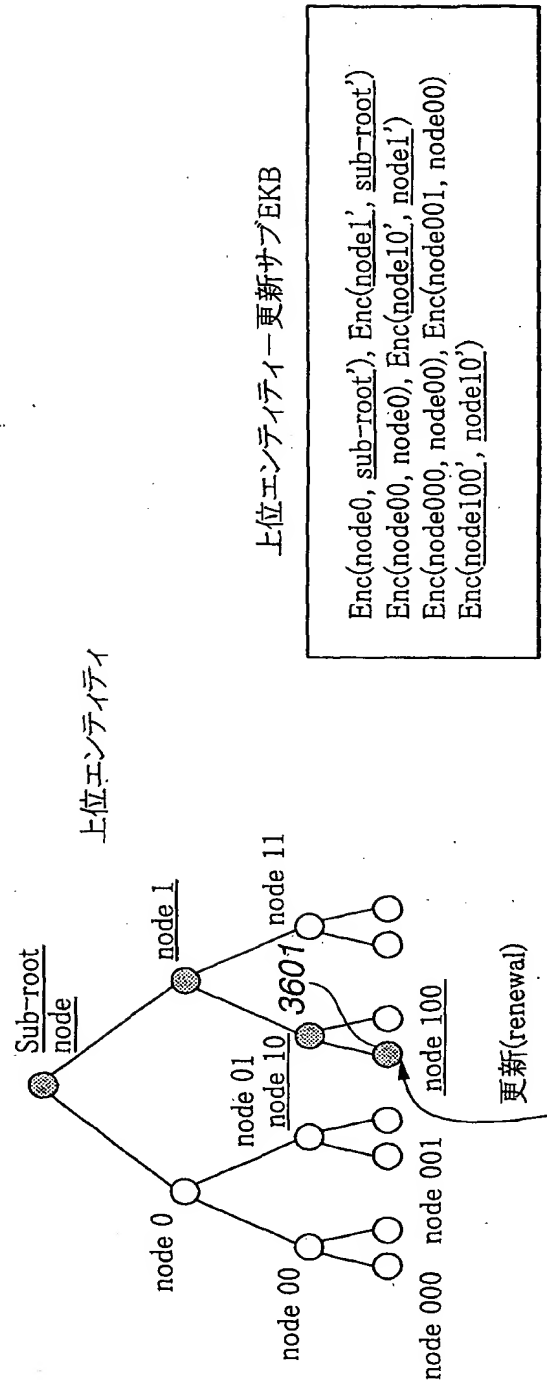


FIG.37B

FIG.37A

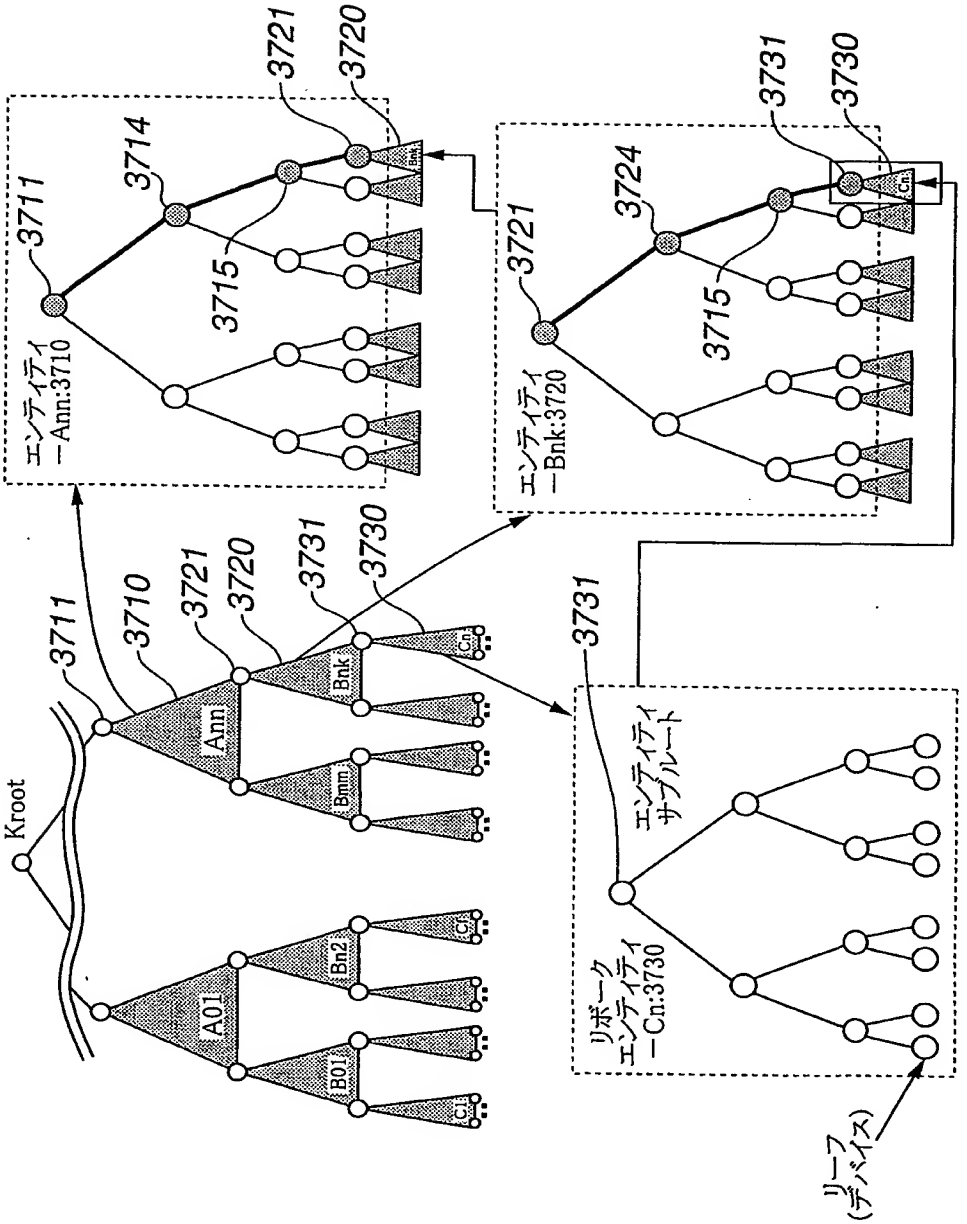


FIG.38A

FIG.38B

FIG.38C

FIG.38D

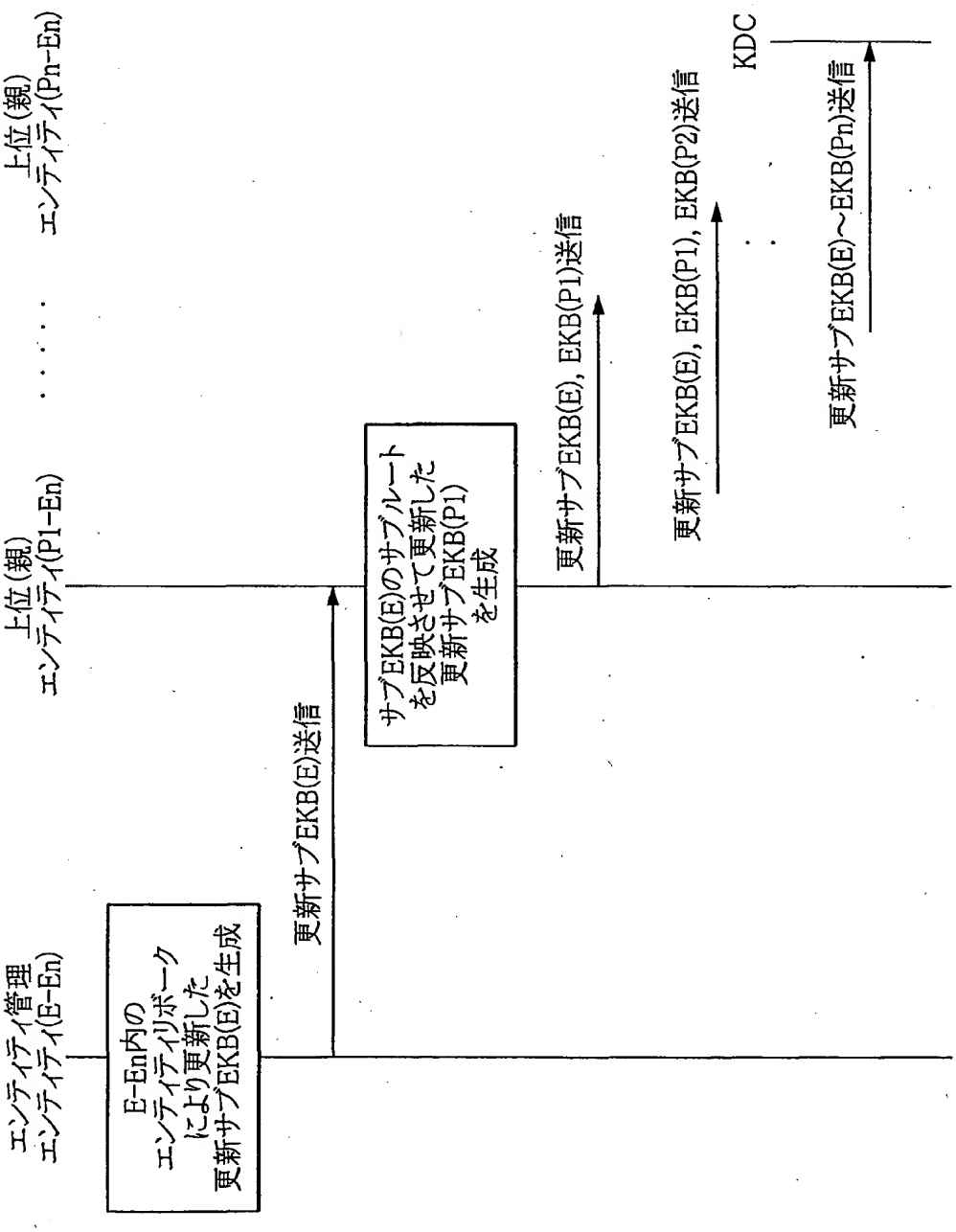

エンティティ管理
エンティティ(E-En)
E-En内の
エンティティリボーク
により更新した
更新サブEKB(E)を生成
更新サブEKB(E)送信
上位(親)
エンティティ(P1-En)
サブEKB(E)のサブルート
を反映させて更新した
更新サブEKB(P1)
を生成
更新サブEKB(E), EKB(P1)送信
上位(親)
エンティティ(Pn-En)
更新サブEKB(E), EKB(P1), EKB(P2)送信
KDC
更新サブEKB(E)～EKB(Pn)送信

FIG.39

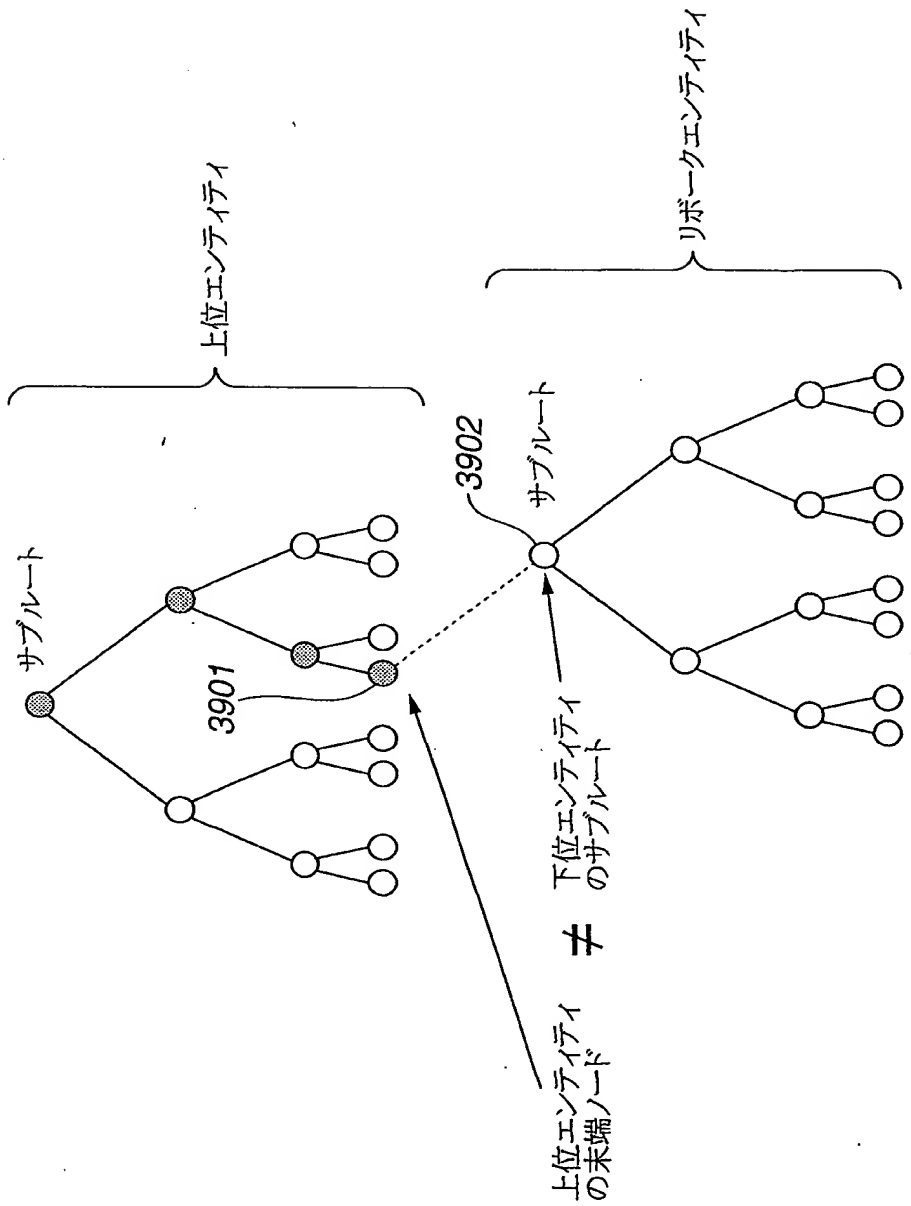

上位エンティティ
リボークエンティティ
サブルート
3901
3902
サブルート
上位エンティティ
の末端ノード
≠
下位エンティティ
のサブルート

FIG.40

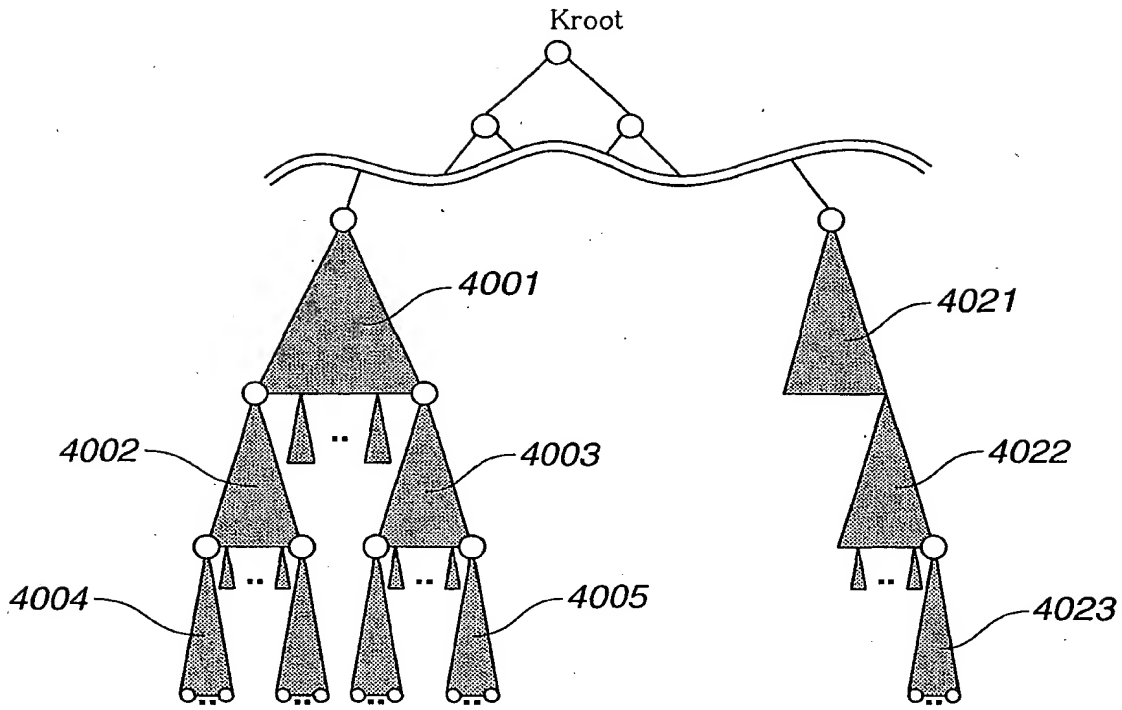


FIG.41

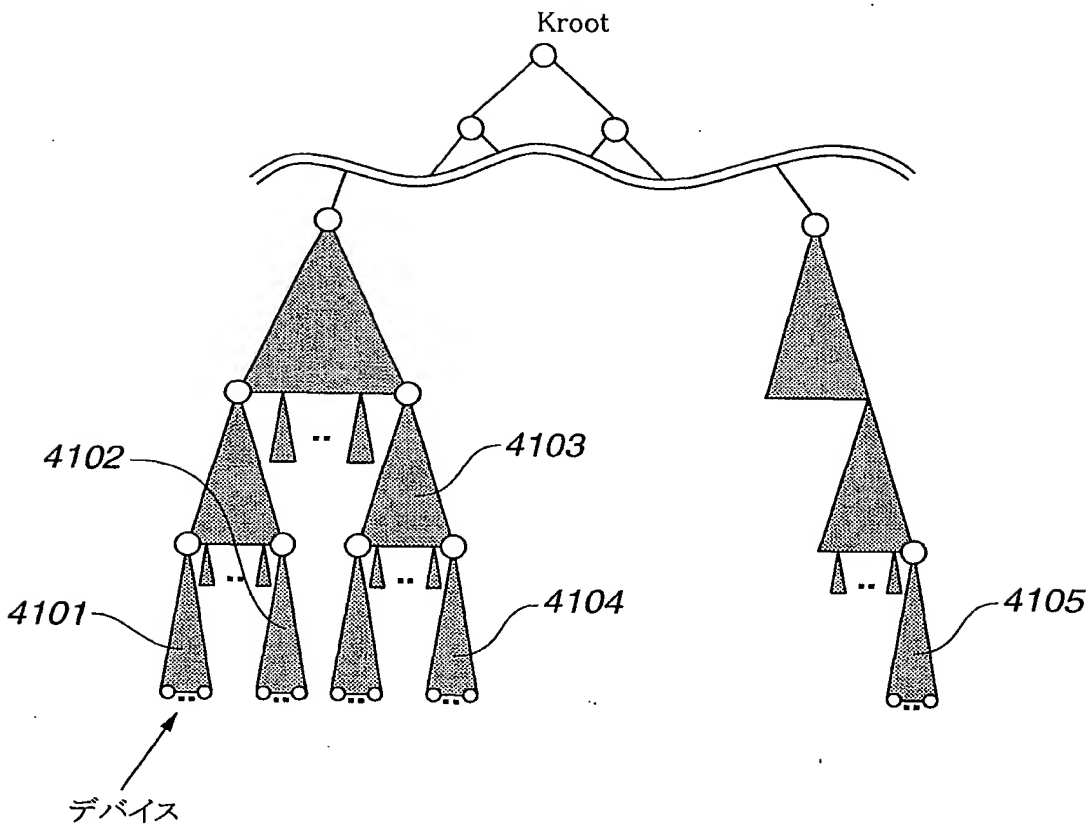
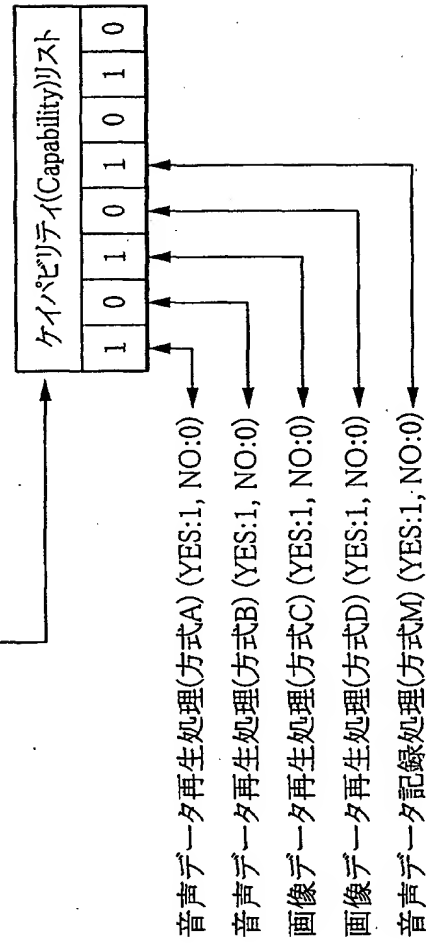


FIG.42

FIG.43A

ケイパビリティ(Capability)リスト	エンティティID	EKB	サブルート
1 0 1 0 1 0 1 0	Ent0010	EKB0010	K0010
0 0 1 0 0 0 1 0	Ent0011	EKB0110	K0110
1 1 1 1 1 0 0 0	Ent0012	EKB0011	K0011
⋮	⋮	⋮	⋮
0 0 1 1 1 1 0 1 1	Ent00nn	EKB00nn	K00nn

FIG.43B



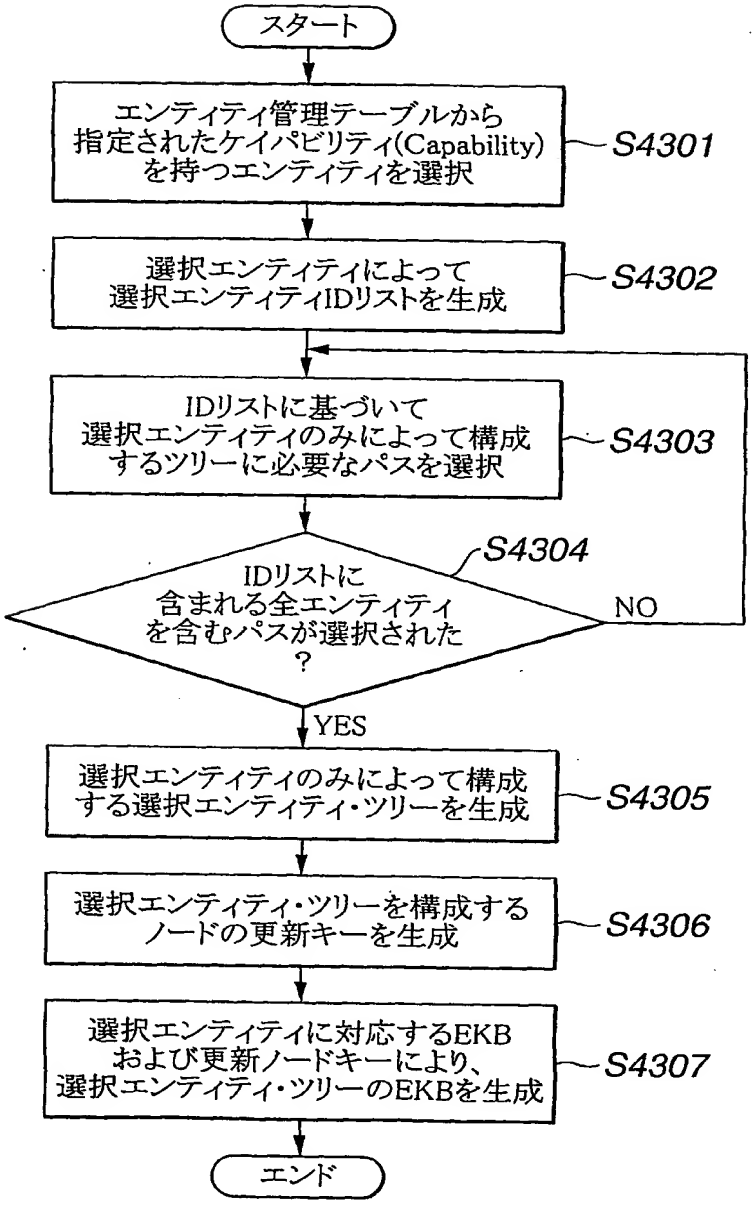

スタート
エンティティ管理テーブルから
指定されたケイパビリティ(Capability)
を持つエンティティを選択
S4301
選択エンティティによって
選択エンティティIDリストを生成
S4302
IDリストに基づいて
選択エンティティのみによって構成
するツリーに必要なパスを選択
S4303
S4304
IDリストに
含まれる全エンティティ
を含むパスが選択された
?
NO
YES
選択エンティティのみによって構成
する選択エンティティ・ツリーを生成
S4305
選択エンティティ・ツリーを構成する
ノードの更新キーを生成
S4306
選択エンティティに対応するEKB
および更新ノードキーにより、
選択エンティティ・ツリーのEKBを生成
S4307
エンド

FIG.44

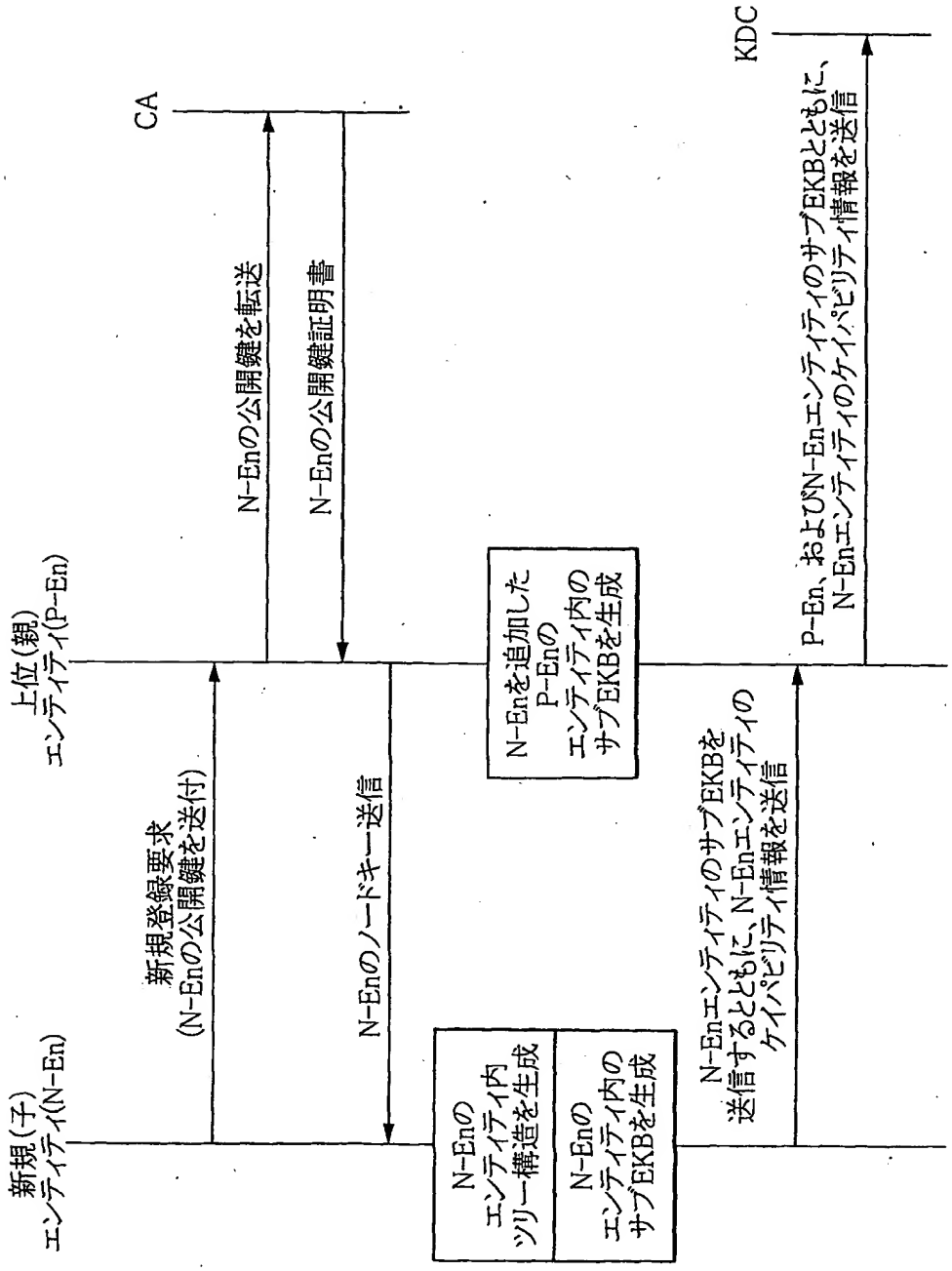

新規(子)
エンティティ(N-En)
上位(親)
エンティティ(P-En)
CA
KDC
新規登録要求
(N-Enの公開鍵を送付)
N-Enの公開鍵を転送
N-Enの公開鍵証明書
N-Enのノードキー送信
N-Enの
エンティティ内
ツリー構造を生成
N-Enの
エンティティ内の
サブEKBを生成
N-Enを追加した
P-Enの
エンティティ内の
サブEKBを生成
N-Enエンティティのサブ EKBを
送信するとともに、N-Enエンティティの
ケイパビリティ情報を送信
P-En、およびN-Enエンティティのサブ EKBとともに、
N-Enエンティティのケイパビリティ情報を送信

FIG.45

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP01/02929

A. CLASSIFICATION OF SUBJECT MATTER
Int.Cl[7] H04L9/00, G06F17/60, G11B20/10, G11B20/12

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
Int.Cl[7] H04L9/00, G06F17/60, G11B20/10, G11B20/12

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Jitsuyo Shinan Koho 1922-1996 Toroku Jitsuyo Shinan Koho 1994-2001
Kokai Jitsuyo Shinan Koho 1971-2001 Jitsuyo Shinan Toroku Koho 1996-2001

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
WPI,JICST DATABASE ON SCIENCE AND TECHNOLOGY key, tree

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JP, 11-187013, A (IBM Japan, Ltd.), 09 July, 1999 (09.07.99), Par. Nos. 9 to 11, 17 to 22	1,8-14,20-25, 28,29,33-39
A	& CN, 1224962, A	2-7,15-19,26, 27,30-32
X	"The VersaKey Framework:Versatile Group Key Management", IEEE Journal on Selected Areas in Communications, Vol.17, No.9, pp.1614-1631	1,8-14,20-24, 28,29,33-39
A	September, 1999 (09.99), page 1616, right column to page 1621, left column	2-7,15-19, 25-27,30-32
X	"Secure Group Communications Using Key Graphs", Proceedings of ACM SIGCOMM'98, pp.68-79 02 September, 1998 (02.09.98), 3.4 Leaving a tree key graph	1,8-14,20-24, 28,29,33-39
A	(http://www.acm.org/sigcomm/sigcomm98/tp/technical.html)	2-7,15-19, 25-27,30-32
X	"Key management for secure Internet Multicast using Boolean Function Minimization Techniques", Proceddings of Infocom'99, Vol.2, p.689-698	1,8-14,20-24, 28,29,33-39

☒ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier document but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
16 May, 2001 (16.05.01)	22 May, 2001 (22.05.01)
Name and mailing address of the ISA/ Japanese Patent Office	Authorized officer
Facsimile No.	Telephone No.

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP01/02929

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	24 March, 1999 (24.03.99) II. KEY MANAGEMENT SCHEME (http://www.ieee-infocom.org/1999/)	2-7,15-19, 25-27,30-32
EX	US, 6049878, A (Sun Microsystems, Inc.), 11 April, 2000 (11.04.00), Full text (Family: none)	1,8-14,20-24, 28,29,33-39
A	US, 5748736, A (S, Mittra), 05 May, 1998 (05.05.98), Full text (Family: none)	1-39
EA	WO, 01/03364, A1 (Matsushita Electric Ind. Co., LTD.), 11 January, 2001 (11.01.01), Full text (Family: none)	1-39
EA	WO, 01/03365, A1 (Matsushita Electric Ind. Co., LTD.), 11 January, 2001 (11.01.01), Full text (Family: none)	1-39

国際調査報告

国際出願番号　PCT／JP01／02929

A.　発明の属する分野の分類（国際特許分類（IPC））

Int. Cl⁷　H04L9／00, G06F17／60, G11B20／10, G11B20／12

B.　調査を行った分野

調査を行った最小限資料（国際特許分類（IPC））

Int. Cl⁷　H04L9／00, G06F17／60, G11B20／10, G11B20／12

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報	1922－1996年
日本国公開実用新案公報	1971－2001年
日本国登録実用新案公報	1994－2001年
日本国実用新案登録公報	1996－2001年

国際調査で使用した電子データベース（データベースの名称、調査に使用した用語）

WPI，JICST科学技術文献データベース　key, tree

C.　関連すると認められる文献

引用文献のカテゴリー＊	引用文献名　及び一部の箇所が関連するときは、その関連する箇所の表示	関連する請求の範囲の番号
X	JP, 11-187013, A（日本アイ・ビー・エム株式会社） 9. 7月. 1999(09.07.99) 第9-11, 17-22段落 & CN, 1224962, A	1, 8-14, 20-25, 28, 29, 33-39
A		2- 7, 15-19, 26, 27, 30-32
X	"The VersaKey Framework:Versatile Group Key Management", IEEE Journal on Selected Areas in Communications, Vol.17, No.9 , p.1614-1631 9月. 1999(09.99) 第1616頁右欄－第1621頁左欄	1, 8-14, 20-24, 28, 29, 33-39
A		2- 7, 15-19,

☒ C欄の続きにも文献が列挙されている。　☐ パテントファミリーに関する別紙を参照。

＊　引用文献のカテゴリー

「A」特に関連のある文献ではなく、一般的技術水準を示すもの
「E」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの
「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献（理由を付す）
「O」口頭による開示、使用、展示等に言及する文献
「P」国際出願日前で、かつ優先権の主張の基礎となる出願の日の後に公表された文献
「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの
「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの
「Y」特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの
「&」同一パテントファミリー文献

国際調査を完了した日 16. 05. 01	国際調査報告の発送日 22.05.01
国際調査機関の名称及びあて先 日本国特許庁（ISA／JP） 郵便番号100－8915 東京都千代田区霞が関三丁目4番3号	特許庁審査官（権限のある職員） 中里　裕正　5M　9364 電話番号　03－3581－1101　内線　3597

様式PCT／ISA／210（第2ページ）（1998年7月）

国際調査報告

国際出願番号 PCT／JP01／02929

C（続き）. 関連すると認められる文献

引用文献のカテゴリー＊	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する請求の範囲の番号
		25-27, 30-32
X	"Secure Group Communications Using Key Graphs", Proceedings of ACM SIGCOMM'98, p.68-79 2. 9月.1998(02.09.98) 3.4 Leaving a tree key graph (http://www.acm.org/sigcomm/sigcomm98/tp/technical.html)	1, 8-14, 20-24, 28, 29, 33-39
A		2- 7, 15-19, 25-27, 30-32
X	"Key management for secure Internet Multicast using Boolean Function Minimization Techniques", Proceddings of Infocom'99, Vol.2, p.689-698 24, 3月.1999(24.03.99) II. KEY MANAGEMENT SCHEME (http://www.ieee-infocom.org/1999/)	1, 8-14, 20-24, 28, 29, 33-39
A		2- 7, 15-19, 25-27, 30-32
EX	US, 6049878, A (Sun Microsystems, Inc.) 11, 4月.2000(11.04.00) 全頁を参照(ファミリ無し)	1, 8-14, 20-24, 28, 29, 33-39
A	US, 5748736, A (S, Mittra) 5, 5月.1998(05.05.98) 全頁を参照(ファミリ無し)	1-39
EA	WO, 01/03364, A1 (Matsushita Electric Industrial Co., LTD.) 11. 1月.2001(11.01.01) 全頁を参照(ファミリ無し)	1-39
EA	WO, 01/03365, A1 (Matsushita Electric Industrial Co., LTD.) 11. 1月.2001(11.01.01) 全頁を参照(ファミリ無し)	1-39